

Министерство образования и науки РФ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Уральский государственный педагогический университет»
Институт математики, физики, информатики и технологий
Кафедра информационно-коммуникационных технологий в образовании

Разработка web-сервиса для оказания бытовых услуг

*Выпускная квалификационная работа
бакалавра по направлению подготовки
09.03.02 – Информационные системы и технологии*

Исполнитель: студент группы БС-51z
Института математики, информатики и ИТ
Исакова Т.А.

Руководитель: к.п.н. старший преподаватель
кафедры ИКТО Арбузов С.С.

Работа допущена к защите
« ____ » _____ 2017 г.
Зав. кафедрой _____

Екатеринбург – 2017

Реферат

Исакова Т.А. РАЗРАБОТКА WEB-СЕРВИСА ДЛЯ ОКАЗАНИЯ БЫТОВЫХ УСЛУГ, выпускная квалификационная работа, 56 стр., рис. 22, приложений 1.

Ключевые слова: РАЗРАБОТКА WEB-СЕРВИСА, WEB-СЕРВИС, БЫТОВЫЕ УСЛУГИ.

Объект разработки – web-сервис для оказания бытовых услуг.

Цель работы – разработать web-сервис для оказания бытовых услуг.

В работе приведены результаты исследования понятий web-сервиса, его специфики и особенности технической реализации, цель которого была достигнута: описать эргономические требования для разработки web-сервиса. Проведенный анализ обосновал выбор средства для реализации, были спроектированы шаблоны страниц web-сервиса для его реализации.

Web-сервис реализован в системе Joomla, состоит из шаблонов, модулей, компонентов, плагинов, расширений и пр.

Оглавление

ВВЕДЕНИЕ.....	4
ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ РАЗРАБОТКИ WEB-СЕРВИСОВ 5	
1.1 ОСНОВНЫЕ ПОНЯТИЯ РАЗРАБОТКИ WEB-СЕРВИСОВ.....	5
1.2 СРЕДСТВА РАЗРАБОТКИ WEB-СЕРВИСОВ.....	14
1.3 ФОРМАЛИЗОВАННОЕ ОПИСАНИЕ ТЕХНИЧЕСКОГО ЗАДАНИЯ.....	23
ГЛАВА 2. ПРАКТИЧЕСКАЯ (ОРИГИНАЛЬНАЯ) ЧАСТЬ.....	30
2.1 МОДЕЛЬНЫЕ ПРЕДСТАВЛЕНИЯ ОБЪЕКТА РАЗРАБОТКИ.....	30
2.2 ОПИСАНИЕ ПРОДУКТА (РЕЗУЛЬТАТА РАЗРАБОТКИ).....	34
2.3 РЕЗУЛЬТАТЫ АПРОБАЦИИ, ТЕХНИЧЕСКАЯ ДОКУМЕНТАЦИЯ.....	41
ЗАКЛЮЧЕНИЕ.....	42
СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ.....	43
ПРИЛОЖЕНИЯ.....	47
Приложение 1.....	47
Приложение 2.....	56

Введение

Сеть Интернет стала общепризнанным фактором деловой и общественной жизни. Широкая распространенность и возросшая пропускная способность создают условия, при которых выгодно решать многие задачи при помощи интернет-технологий.

Однако Интернет объединяет в себе много различных платформ, а информация содержится в разнообразных источниках данных. Поэтому актуальна проблема связи таких разнородных данных, а также создания способа, который позволяет получать их в виде удобном для дальнейшей обработки.

Концепция веб-сервисов (Web Services) призвана решить эту задачу объединения, интеграции разнородных систем на основе открытых стандартов.

На сегодняшний день огромное количество людей пользуется услугами web-сервисов.

Предмет разработки: web-сервис для оказания бытовых услуг.

Цель: спроектировать и разработать web-сервис для оказания бытовых услуг.

Для реализации поставленной цели необходимо решить следующие задачи:

1. Изучить основные понятия разработки web-сервисов;
2. Рассмотреть самые популярные и востребованные CMS платформы и языки для разработки web-сервисов;
3. Выбрать и изучить систему для разработки web-сервиса;
4. Спроектировать и разработать web-сервис для оказания бытовых услуг;
5. Провести тестирование и апробацию продукта.

Глава 1. Теоретические основы разработки web-сервисов

1.1 Основные понятия разработки web-сервисов

Интернет сегодня – это то, без чего не может жить подавляющая часть населения. Данная технология связывает людей со всего мира, с разных уголков земли. Очень трудно найти человека, который не пользовался хоть раз Интернетом, еще сложнее найти человека, который о нем не слышал.

Для кого-то Интернет – это просто неиссякаемый источник информации, для кого-то – это работа, здесь человек трудится и зарабатывает себе на жизнь. А для кого-то Интернет – это среда обитания, неотъемлемая часть его жизни, без которой он уже не представляет своего существования.

Интернет настолько удобен для обмена информацией, что просто не мог не завоевать ту популярность, которой он обладает сегодня. При разработке современного программного обеспечения для Интернета все больше вниманию уделяется разработке такого типа приложений, как web-сервисы.

В развитых странах, Интернет уже давно доминирует в области социального обслуживания: при помощи использования web-сервисов можно покупать билеты на поезда и самолеты, оплачивать коммунальные услуги, приобретать бытовую технику. Это избавляет население от пустой траты времени стоя в очередях, поисках магазинов. В связи со сказанным, можно считать, что разработка и применение web-сервисов для оказания социальных услуг, в том числе бытовых, является весьма актуальной темой в современном обществе.

Существует достаточно много определений понятия «web-сервис» [1, 7, 26, 28, 29]. Но чаще всего web-сервис определяется как «web-ресурс со стандартизированными интерфейсами, который может взаимодействовать с другими программными модулями посредством сообщений, основанных на определенных стандартах и протоколах» [26, с. 4].

Главным отличием web-сервиса от web-сайта является, то, что web-сервис может предоставлять определенные услуги. Например, такой web-сервис как интернет-магазин предлагает пользователю определенный товар, который он может приобрести непосредственно на данном web-сервисе. Web-сайт – это сайт на котором располагается какая-либо информация, с которой пользователь может ознакомиться.

Анализ работ посвященных разработке web-сервисов позволяет выделить два наиболее важных направления: дизайн и навигация. Дизайн играет важную роль в привлечении внимания пользователей сети Интернет к web-сервису. Навигация в web-сервисе должна быть простой и понятной – это положительно сказывается на конверсии и даже на ранжировании в поисковиках.

Технологии создания web-сервисов, базирующиеся на возможностях Интернет, призваны кардинально улучшить взаимодействие людей и информационных систем друг с другом и обеспечить взаимное проникновение различных систем и процессов. Они образованы из целого ряда стандартных протоколов взаимодействия, средств описания моделей данных и интерфейсов, а также вспомогательных сетевых служб, обеспечивающих доступность бизнес-функций организаций авторизованным пользователям через Интернет с любого подключенного к нему устройства.

Существует множество платформ и языков, на которых может быть разработан web-сервис. Выделяют несколько этапов разработки web-сервиса: планирование, реализация, тестирование, публикация, рекламирование, сопровождение, рассмотрим эти этапы подробно [16, 34]:

- **Планирование.** На этом этапе должны быть получены ответы на следующие вопросы: цель создания web-сервиса, аудитория, на которую он рассчитан, какая информация будет размещена. Целесообразно познакомиться с сайтами, тематика которых близка к тематике создаваемого web-сервиса, с тем, чтобы избежать ошибок или поучиться на положительных примерах. Определяются категории, подразделы сайта, количество страниц в каждом разделе, подразделе. Разрабатывается структурная (навигационная) схема сайта.
- **Реализация.** На этом этапе проводится подготовка текстового и графического материала (печать, сканирование, оптимизация). Материал разбивается по файлам в соответствии со структурой. Организовываются ссылки между web-страницами сайта. Разрабатывается дизайн. На этом этапе определяются: фон, на котором будет расположена информация, цвет текста, цвет гипертекстовых ссылок, знаки навигации, выделение разделов графикой или текстовыми ссылками, представление фотографического материала.
- **Тестирование.** После того как web-сервис создан, необходимо проверить правильность переходов со страницы на страницу. В связи с тем, что браузеры разных фирм иногда по-разному интерпретируют встречающиеся в HTML-документе команды разметки, а браузеры устаревших версий «не понимают» описаний, соответствующих новым стандартам, желательно просмотреть web-сервис в браузерах разных производителей и разных версий, при разных разрешениях экрана, добиваясь приемлемого результата.
- **Публикация.** Размещение сайта в Internet. Для этого необходимо разместить все файлы web-сервиса на web-сервере, предоставляющем такие услуги.
- **Рекламирование.** На этом этапе предполагается взаимный обмен ссылками с другими сайтами, схожими по тематике, включение адреса сайта в письма, визитные карточки, брошюры, буклеты и т.д., регистрация сайта в популярных поисковых серверах;

- **Сопровождение.** После создания web-сервиса нужно не забывать обновлять информацию, расширять материал, улучшать дизайн, в противном случае велика вероятность того, что потенциальные посетители web-сервиса не будут к нему возвращаться [34].

Из курса, представленного А.В. Сычевым на тему «Перспективные технологии и языки веб-разработки» и «Web-технологии» [10, 11] я хочу выделить следующее: web-сервис является единицей модульности при использовании сервис-ориентированной архитектуры приложения. Сервис-ориентированная архитектура (англ. SOA, service-oriented architecture) – представляет собой модульный подход к разработке программного обеспечения, основанный на использовании сервисов со стандартизированными интерфейсами. В основе SOA лежат принципы многократного использования функциональных элементов информационных технологий, унификации типовых операционных процессов. Компоненты программы могут быть распределены по разным узлам сети, и предлагаются как независимые и слабо связанные, заменяемые сервисы-приложения. Интерфейс компонентов SOA-программы осуществляет инкапсуляцию деталей реализации конкретного компонента (ОС, языка программирования и т.п.). SOA хорошо зарекомендовала себя при построении крупных корпоративных программных систем. Целый ряд разработчиков и интеграторов предлагают инструменты и решения на основе SOA (например, платформы Microsoft .NET, IBM Web Sphere, SAP Net Weaver, Diasoft и др.).

Структура реализации web-сервиса, заключающая в себе программные элементы, их внешние свойства и взаимосвязи, называется архитектурой программы. Важность программной архитектуры объясняется тем, что в ней излагаются начальные проектные решения, она определяет ограничения реализации и организационную структуру разрабатываемой системы, кроме того, программная архитектура позволяет более точно планировать ресурсы, необходимые для создания программного проекта. Программная архитектура содержит в себе структуры, из которых складываются крупные программные системы. Архитектурное представление системы абстрактно; не затрагивает детали реализации, алгоритмы и представления данных, оно ориентировано на поведение и взаимодействие «черных ящиков». На архитектуру программного обеспечения оказывают влияние множество факторов, в том числе [13, 32]:

- заинтересованные в системе лица, такие как разработчики, заказчики;
- компания-разработчик с выработанными в ней подходами к реализации программных проектов;
- опыт и привычки архитекторов программной системы;
- современная разработка, техническая база, включающая программные и аппаратные ресурсы разработки программного обеспечения.

Рассмотрим подробно архитектуру web-сервисов. Архитектура программной системы охватывает не только ее структурные и поведенческие аспекты, но и правила ее использования и интеграции с другими системами, функциональность, производительность, гибкость, надежность, возможность повторного применения, полноту, экономические и технологические ограничения, а также вопрос пользовательского интерфейса. По мере развития информационных технологий и программных систем все большее значение приобретает их интеграция друг с другом с целью построения единого информационного пространства. Как можно видеть из вышеприведенных определений интеграция является важнейшим элементом архитектуры. Для того чтобы построить правильную и надежную архитектуру и грамотно спроектировать совместную работу программных систем необходимо четко следовать современным стандартам в области информационно-коммуникационных технологий. Без этого велика вероятность, создать архитектуру, которая неспособна развиваться и удовлетворять растущим потребностям пользователей в сети Интернет. В качестве законодателей стандартов в области информационно-коммуникационных технологий выступают такие международные организации как SEI (Software Engineering Institute), WWW (консорциум World Wide Web), OMG (Object Management Group), организация разработчиков Java – JCP (Java Community Process), IEEE (Institute of Electrical and Electronics Engineers) и другие. Web-сервисы представляют собой приложения с сервис-ориентированной архитектурой (Service-Oriented Architecture – SOA) [2, 27]. SOA представляет собой модульный подход к разработке программного обеспечения, основанный на использовании сервисов (служб) со стандартизированными интерфейсами [10]. В настоящее время наблюдается устойчивый рост интереса к концепции сервис-ориентированной архитектуры (Service-Oriented Architecture – SOA). SOA [35] – это архитектурная модель для объединения доступных вычислительных ресурсов, таких как: приложения, данные, формы их

представления, средства управления и мониторинга, с требованиями достижения желаемых результатов, для потребителей сервисов, которыми могут быть: конечные пользователи, другие приложения, или другие, подобные, сервисы. В общем виде SOA [10, 32] предполагает наличие трех основных участников: поставщика сервиса, потребителя сервиса и реестра сервисов (Рис. 1).

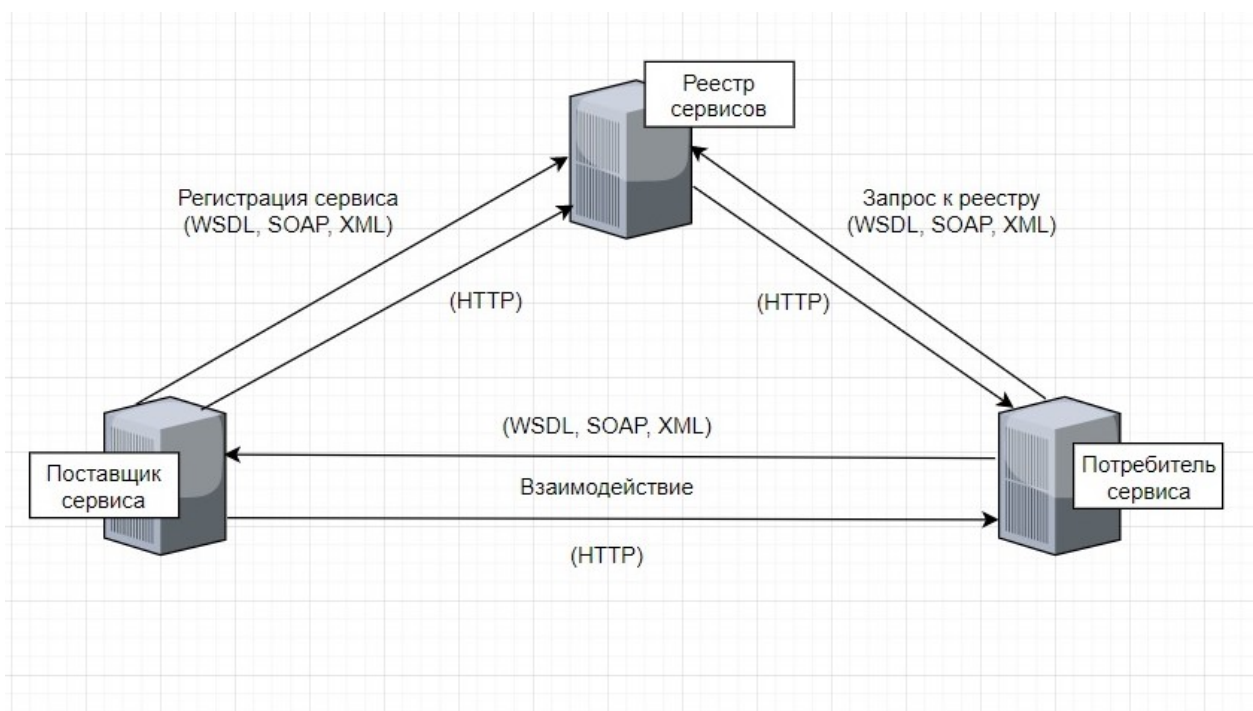


Рис. 1 Общая схема сервис-ориентированной архитектуры

Взаимодействие участников выглядит достаточно просто: поставщик сервиса регистрирует свои сервисы в реестре, а потребитель обращается к сервису с запросом.

Организация по стандартизации OASIS [35] утвердила эталонную модель сервис-ориентированной архитектуры Reference Model for Service-Oriented Architecture 1.0 (SOA-RM). Модель призвана ввести четкую техническую терминологию SOA для разработчиков и архитекторов. Работавший над моделью технический комитет OASIS определяет SOA-RM как абстрактную базу для понимания основных объектов и взаимодействия между ними в сервис-ориентированной среде и для разработки согласованных стандартов и спецификаций, поддерживающих данную среду. Модель унифицирует концепции SOA и может быть использована архитекторами для разработки SOA или в обучении SOA. В комитете отмечают также, что SOA-RM не связана непосредственно ни с какими стандартами, технологиями или другими деталями конкретных реализаций. Цель модели – предоставить общую схему, которая снимет все двусмысленности в различных реализациях SOA. Эталонная модель от OASIS поможет архитекторам координировать отдельные проекты по SOA в организации или планировать создание единой корпоративной архитектуры, но абстрактность ее концепций препятствует применению SOA-RM в конкретных реализациях сервис-ориентированной архитектуры.

Работа web-сервисов построена на использовании нескольких открытых стандартов [23]: XML – расширяемый язык разметки, предназначенный для хранения и передачи структурированных данных; SOAP – протокол обмена сообщениями на базе XML; WSDL – язык описания внешних интерфейсов веб-сервисов на базе XML; UDDI – универсальный интерфейс распознавания, описания и интеграции (Universal Discovery, Description, and Integration). Каталог веб-сервисов и сведений о компаниях, предоставляющих веб-сервисы во всеобщее пользование или конкретным компаниям.

Связь между web-сервисами и их клиентами осуществляется посредством сообщений в формате XML. SOAP (Simple Object Access Protocol - простой протокол доступа к объектам) представляет собой протокол сообщений для выбора web-сервисов. SOAP [4, 5, 6, 10] основан на языке XML и расширяет некоторые протокол прикладного уровня: HTTP, FTP, SMTP и т.д. Как правило, чаще всего используется HTTP. Вместо использования HTTP для запроса HTML-страницы, которая будет показана в браузере, SOAP отправляет посредством HTTP-запроса XML-сообщение и получает результат в HTTP-ответе. Для правильной обработки XML-сообщения процесс-«слушатель» HTTP (напр. Apache или Microsoft IIS) должен предоставить SOAP-процессор, или, другими словами, должен иметь возможность обрабатывать XML. SOAP [5, 10] является самой главной частью технологии web-сервисов. Он осуществляет перенос данных по сети из одного места в другое. SOAP обеспечивает доставку данных web-сервисов. Он позволяет отправителю и получателю XML-документов поддерживать общий протокол передачи данных, что обеспечивает эффективность сетевой связи.

SOAP – это базовая однонаправленная модель соединения, обеспечивающая согласованную передачу сообщения от отправителя к получателю, потенциально допускающая наличие посредников, которые могут обрабатывать часть сообщения или добавлять к нему дополнительные элементы. Спецификация SOAP содержит соглашения по преобразованию однонаправленного обмена сообщениями в соответствии с принципом «запрос/ответ», а также определяет, как осуществлять передачу всего XML-документа. Каждый web-сервис предоставляет документ WSDL (Web Service Description Language – язык описания web-сервиса), в котором описывается все, что клиенту необходимо для работы с этим сервисом.

WSDL-документ [10] предоставляет простой и последовательный способ задания разработчиком синтаксиса вызова любого web-метода. Более того, этот документ позволяет использовать инструменты автоматического генерирования прокси-классов, подобные включенным в среды Visual Studio .NET и .NET Framework. Благодаря указанным средствам использование web-сервиса является таким же простым, как и применение локального класса. WSDL-документ имеет основанный на XML формат, в соответствии с которым информация подразделяется на пять групп. Первые три группы представляют собой абстрактные определения, не зависящие от особенностей платформы, сети или языка, а оставшиеся две группы включают конкретные описания.

Спецификация UDDI [11] (Universal Description, Discovery, and Integration - универсальное описание, поиск и интеграция) позволяет избежать различных проблем посредством использования специального хранилища (репозитория), где предприятия и организации могут размещать данные о предоставляемых ими сервисах, включая Oracle и Microsoft. Объединив свои усилия, эти компании разработали проект спецификации UDDI, которая по истечении 18 месяцев была стандартизирована. Информация в этом репозитории должна обновляться вручную. С этой целью некоторые «узловые операторы» хранят идентичные копии репозитория UDDI. Эти компании обеспечивают хранение указанного репозитория и бесплатный доступ к нему для популяризации web-сервисов. Кроме того, Майкрософт включила версию UDDI в программное обеспечение сервера Windows .NET для использования в корпоративных сетях интранета. В хранилище UDDI содержатся сведения о предприятиях, предоставляющих web-сервисы, о типе каждого сервиса и связях с информацией и спецификациями, относящимися к этим сервисам. Интерфейс UDDI сам по себе представляет собой web-сервис. Для регистрации или поиска службы следует отправить SOAP-сообщение.

Главными недостатками web-сервисов являются меньшая производительность и больший размер сетевого трафика по сравнению с такими технологиями как RMI, CORBA, DCOM за счет использования текстовых XML-сообщений. В основе SOA [10, 11] лежат принципы многократного использования функциональных элементов информационных технологий, ликвидации дублирования функциональности в программном обеспечении, унификации типовых операционных процессов. Компоненты программы могут быть распределены по разным узлам сети, и предлагаются как независимые, слабо связанные, заменяемые сервисы-приложения. Интерфейс компонентов SOA-программы предоставляет изолирование (инкапсуляцию) деталей реализации конкретного компонента (операционной системы, платформы, языка программирования, вендора, и т. п.) от остальных компонентов.

Таким образом, SOA предоставляет гибкий и элегантный способ комбинирования и многократного использования компонентов для построения сложных распределённых программных комплексов. Сервис-ориентированная архитектура позволяет достичь следующие цели: сокращение издержек при разработке приложений, за счёт упорядочивания процесса разработки, расширение повторного использования кода, независимость от используемых платформ, инструментов, языков разработки, повышение масштабируемости создаваемых систем, улучшение управляемости создаваемых систем. Взаимодействие компонентов в данной архитектуре происходит благодаря использованию протокола HTTP (HyperText Transfer Protocol) [2, 11]. Он представляет собой протокол прикладного уровня для передачи гипертекста. Является протоколом высокого уровня, использующийся в сетях TCP/IP для извлечения файлов с соответствующих ресурсов. При этом HTTP не хранит информацию о своем состоянии: каждая транзакция рассматривается независимо от остальных. Следовательно, в типичной реализации создается новое TCP-соединение между клиентом и сервером для каждой транзакции, которое прерывается сразу после ее завершения.

Таким образом, при разработке web-сервисов основными принципами являются: независимость от платформы и языка программирования, простота реализации, поиска web-сервисов и интеграции приложений с их помощью. Наиболее популярные и востребованные языки программирования и платформы рассмотрим в параграфе 1.2.

1.2 Средства разработки web-сервисов

Как уже было сказано ранее, существует множество платформ и языков программирования для создания web-сервисов, рассмотрим некоторые из них.

Язык программирования – формальная знаковая система, предназначенная для записи компьютерных программ. Язык программирования определяет набор лексических, синтаксических и семантических правил, задающих внешний вид программы и действия, которые выполнит исполнитель (компьютер) под её управлением [24].

Существует рейтинг самых популярных и востребованных языков программирования. В 9-ку наиболее популярных языков программирования на данный момент входят следующие языки [12]:

1. Java.
2. C.
3. C++.
4. Python.
5. C#.
6. Visual Basic .NET.
7. Javascript.
8. PHP.
9. Perl.

Рассмотрим особенности данных языков программирования подробнее.

Java [17]. Язык программирования Java является одним из самых востребовательных в последние несколько лет, и эта тенденция сохраняется и на 2017 год. Java – типизированный объектно-ориентированный язык, получивший наибольшую известность под развитием компании Oracle, которая, по правде, приобрела наработки данного шедевра у родительской компании под названием Sun Microsystems. Первый официальный выпуск языка состоялся в уже далеком 1995 году. С тех пор это самый популярный язык и вот почему:

Данный язык используется уже десятилетия. Он постоянно совершенствовался все это время и сейчас в нем присутствует просто масса различных продвинутых технологий и решений;

Освоив Джава, можно легко разобраться практически в любом другом языке, том же C++ и полноценно использовать в своей работе уже несколько инструментов;

Java довольно прост и чуть-ли не интуитивно понятен. Достаточно изучить основы и хоть немного разбираться в английском, чтобы быстро сориентироваться в нем и начать писать код;

Синтаксис Джава является образцом минимализма и своеобразной «красоты». С его помощью можно наглядно понимать принципы самого программирования. Это используется многими авторами в учебных целях [12].

Язык C. Си является типизированным языком, разработки которого начались еще в 60-х годах 20-го века. Изначально он был разработан для создания ОС Unix, но по прошествии некоторого времени приобрел популярность и стал использоваться другими системами.

Именно язык C стал тем базисом, на котором в будущем развивались C++, Java и другие. На нем была написана масса полезных приложений. Главными его преимуществами, которые о сих пор держат C в лидерах среди языков программирования, называют следующее:

- Простота и возможность легко выражать любые идеи. Код языка довольно понятен – здесь нет классов, а используемая система типов вполне доступна для понимания;
- Написанное на C ПО работает очень быстро. Это касается и отдельных процессов, и всей программы в целом;
- На C можно найти массу помощников, отладчиков и аварийных дампов, которые существенно облегчают работу с кодом.

И это только некоторые его плюсы.

C++. C++ – современный востребованный язык программирования, с помощью которого пишется львиная доля всего программного обеспечения на компьютерные операционные системы. Это типизированный язык общего назначения, посредством которого можно реализовать объектно-ориентированное, процедурное, обобщенное программирование. В C++ собраны достоинства многих других языков, к примеру, он одновременно сочетает в себе особенности как низкоуровневых, так и высокоуровневых языков.

На C++ пишутся не только программы для компьютеров, но и масса других вещей: драйвера для различного оборудования, ПО для серверов, встраиваемых систем. Множество платформ работают преимущественно с данным языком. В 2017 году он также популярный и перспективный – хорошо зная C++ в наше время, можно найти себе работу практически всегда и в любой стране.

Python. Python является высокоуровневым языком, главная задача, которая перед ним изначально ставилась – дать разработчику максимальную производительности труда при минимуме возникающих проблем и хорошая читаемость получаемого кода. Используемый в Питон синтаксис предельно упрощен и минимизирован, тем не менее, его вполне достаточно для реализации практически любых задач. Имеющаяся объемная стандартная библиотека для Python включает в себя массу полезнейших функций.

Сильные стороны Питона:

- Наличие массы фреймворков, готовых решений и функций;
- Отличный и понятный синтаксис;
- Высокие зарплаты программистов на Python;
- Отлично подходит для реализации различных математических вычислений, создания системных инструментов и прочего.

C#. Язык Си Шарп (C#) разработан в начале 2000-х годов в глубинах компании Microsoft. Главная его задача изначально – создание приложений, оптимизированных под Microsoft .NET Framework. Синтаксис Си Шарп очень близок к лидерам списка: C++ и Java. Имеет массу удобных функций: анонимные функции, итераторы, события, свойства, исключения, комментарии и массу других полезных вещей.

Главные его сильные стороны это:

- Высокая скорость разработки;
- Широкие возможности оптимизации написанного кода и другие.

Visual Basic.NET. Шестое место их списка востребованных языков программирования в 2017 – Visual Basic .NET. Он также, как и предыдущие варианты является объектно-ориентированным и главная сфера его применения – разработки под Microsoft .NET. В этом же выражается и его популярность – так как ОС Windows все еще является лидером на рынке ПО для компьютеров и ноутбуков, программирование под нее посредством Visual Basic .NET имеет большие перспективы даже сейчас, в 2017 году.

JavaScript. В отличии от Java, javascript имеет более узкое применение, но, тем не менее, он очень популярен и в наше время в том числе. Этот язык относится к ООП, является мультипарадигменным, поддерживает работу в императивном и функциональном стилях. Главное его применение пришлось на сферу веб-дизайна – для реализации каких-либо функций браузеров, скриптов и интернет сайтах, создание интерактивных элементов веб-страниц и тому подобных задач. Некоторые новички говорят, что это самый сложный язык программирования, но подобные заявления совершенно не соответствуют действительности.

Еще недавно JavaScript был очень востребованным и специалистам в этой сфере платили большие деньги. Сейчас он постепенно уходит в прошлое, так как все больше для интернета применяется HTML 5.

PHP. Отличный язык, широко применяющийся в сфере веб-технологий. Скриптовый язык, посредством которого разрабатываются самые разнообразные приложения, в основном для работы серверов, динамических страниц сайтов. PHP очень хорош тем, что в его помощь довольно просто работать с базами данных, что очень полезно при обработке больших массивов однотипной информации. Многие администраторы серьезных ресурсов используют в своей деятельности в основном именно PHP.

Perl. Это высокоуровневый и интерпретируемый язык, главное преимущество которого – отличная работа с текстовой информацией. Имеет много сходств с C, оснащен массой дополнительных модулей, которые позволяют использовать Perl в веб-технологиях, программировании ПО и игр, в системном администрировании, и даже при разработке графических интерфейсов. Худшим языком программирования из указанных выше его не стоит называть – просто он имеет немного меньшее значение, нежели предыдущие гиганты [31].

В качестве заключения стоит отметить, что указанные языки программирования все еще актуальны в текущем 2017 году и используются для реализации самых разнообразных задач в разработке ПО, в веб-технологиях, в других сферах. Какой из указанных языков самый перспективный сказать очень сложно. Неплохие шансы есть именно у Java, ведь область мобильного ПО все еще бурно развивается, а также у C и C++. Конечно, если быть хорошим специалистом, то можно найти неплохой заработок, зная и другие указанные выше языки.

Существует три варианта построения web-сервиса. Первый, написать собственные программы, создающие шаблоны и реализующие необходимые функции администрирования. Второй, воспользоваться помощью сторонних разработчиков, что бы написали сайт «под ключ». И, наконец, воспользоваться готовым решением. В интернете существуют системы управления содержимым web-сервиса (CMS), позволяющие реализовывать компромиссное решение между низкой стоимостью статических web-сервисов и высокой гибкостью динамических. Благодаря подобным системам резко повысилась управляемость web-сервисом, и значительно снизились расходы на администрирование web-сервиса [30].

CMS – это система, которая позволяет в удобном виде создавать и управлять (редактировать, удалять) текстовыми материалами и мультимедиа документами (содержимое или контент) на web-сервиса. Аббревиатура «CMS» появилась от англ. Фразы Content Management System, что и переводится как система управления контентом [24].

CMS – это движок на котором создается Веб-сервис. Принцип работы любого движка прост. Пользователь системы добавляет контент на web-сервис. Вся информация, которую ввел пользователь, сохраняется в базе данных или файлах. Когда посетитель заходит на web-сервис, информация читается из базы данных и отображается на web-сервисе. Вид отображения информации зависит от шаблона [24, 29].

Шаблон web-сервиса — это заготовка дизайна web-сервиса, без наполнения её информацией. Почти во всех CMS шаблоны web-сервиса легко меняются [24].

Во многих движках есть система модулей. То есть, функционал системы можно расширить, подключая дополнительные модули. Например, модуль «Чат» или модуль «Обратная связь» и т.д. Модули часто называют плагинами, расширениями или дополнениями.

Пишутся CMS чаще всего на одном из серверных языков программирования (PHP, Perl и др.).

С момента появления интернета в мире было разработано множество различных CMS. Наиболее известных их них, более 100, а если считать еще и CMS, разработанные отдельными программистами или агентствами, то их число может перевалить за десять тысяч. Все они делятся на три основных типа:

- платные CMS;
- бесплатные CMS;
- самописные CMS (условно-бесплатные).

Платные CMS – немногочисленные и не самые популярные системы, ввиду их стоимости. В нашей стране наиболее известной платной CMS является 1С-Битрикс (как в России, так и в мире). Второй по популярности в России является UMI. CMS. Данные системы используются в основном для создания интернет-магазинов.

Бесплатные CMS – наиболее распространенные. Среди самых известных – WordPress, Joomla, Drupal, Blogger, OpenCart и другие. Эти CMS имеют огромную популярность во всем мире и, в том числе, в России. Сферы их использования самые разнообразные, – от простых одностраничных web-сервисов, до сложных интернет-магазинов.

Самописные CMS – самые многочисленные. Это системы, которые разрабатывают «под себя» интернет-агентства, студии и отдельные программисты. Среди них встречаются интересные решения, но очень часто идея таких систем, состоит в том, чтобы привязать заказчика web-сервиса к исполнителю. Самописные CMS можно назвать условно-бесплатными, т.к. заказчик не платит за их использование, но при этом, в будущем, он волею-неволей вынужден сотрудничать с той компанией, которая разработала систему для его web-сервиса [33].

Рассмотрим несколько примеров CMS:

Вордпресс (Wordpress) [18]. Wordpress хорошо русифицирован, для него написано множество дополнительных модулей и сделано множество шаблонов. Конечно, возможно и самим внести изменения в дизайн.

Минусы Wordpress типичны для популярных CMS - не слишком быстрая работа сайта, возможность сбоев при высокой посещаемости и периодическое обнаружение тех или иных дыр в скрипте. Поэтому некоторые пользователи даже при создании блогов отдают предпочтение всё же более простым и лёгким движкам. Также, очевидно, Wordpress вряд ли подойдёт для сложного сайта с большой функциональностью, портала, интернет-магазина и т. п. - для них нужно обратить внимание на более универсальные CMS.

Друпал (Drupal) [14]. Эта CMS подойдёт для создания форумов, блогов (в т. ч. многопользовательских), онлайн-энциклопедий, сайтов сообществ. Однако вряд ли данную систему можно назвать универсальной.

Сравнивая Drupal с Wordpress можно однозначно сказать следующее – эта CMS не для тех, кто хочет создать сайт легко и быстро, установив и сразу же получив готовое. Drupal удобнее для тех, кто готов сидеть и тщательно выстраивать структуру сайта, чтобы получить то, что ему нужно, но не для привыкших к простым решениям новичков. Эта особенность отпугивает от Drupal тех, кто не хочет заморачиваться со всей этой сложностью, но привлекает тех, кто любит гибкость и любит поэкспериментировать.

Web-сервер Apache [19]. Этот Web-сервер является самым распространенным в мире среди серверов. Причин такой популярности много. Прежде всего, это возможность свободно получить его как с основного сервера проекта Apache, так и с «зеркал», расположенных во многих странах мира, в том числе и России. Имеется подробная документация по настройке и администрированию, включая FAQ. В рамках данного проекта ведется подробный учет и исправление найденных ошибок, чему посвящено несколько страниц сервера. Многие разработчики модифицируют код Apache, внося дополнительные функции, и предлагают для свободного распространения свои разработки. В частности, имеются версии Apache, в которые добавлены функции по работе с русскоязычными документами с учетом различных кодировок кириллицы.

Russian Apache это программный продукт, за основу которого был взят популярный HTTP-сервер Apache. К нему была добавлена функциональность, необходимая для корректной поддержки нескольких кодировок кириллицы одновременно, что потребовало внесения добавлений в основной код Apache.

Последней версией Apache является 2.2.x. Эта версия Apache является основной, в новой ветви проекта, и представляет собой наилучшую доступную, стабильную версию HTTP сервера Apache.

Wix.com [9]. Wix — это бесплатный конструктор сайтов, которым пользуются миллионы людей по всему миру. Возможность создать сайт самостоятельно, без специальных знаний и опыта. Никакого программирования и полная свобода творчества. В конструкторе сайтов присутствуют следующие функциональные возможности:

- Наличие шаблонов;
- Возможность просмотра примеров сайтов (примеры уникальных HTML сайтов, созданных пользователями);
- Большое кол-во преимуществ;
- возможность использовать готовые сайты;

- Выбор премиум-планов (неограниченное количество страниц и хостинг);
- Возможность создать интернет-магазин;
- Подключение appMarker (популярные приложения и сервисы);
- Использование доменов.

Джумла (Joomla) [8, 21]. Ещё одна популярная CMS, на ней часто делают сайты. Несколько сложнее в освоении, чем Wordpress, но имеет и большую сферу применения. Для Joomla разработано огромное число модулей, включая форумы, чаты, блоги, интернет-магазины и т. д., поэтому на ней можно смело пытаться делать сложный многофункциональный сайт. Вообще, на Joomla можно делать самые различные сайты, как простые, так и сложные, это универсальная CMS. Также для Joomla существует огромное число шаблонов, и в возможности создания или выбора дизайна для сайта веб-мастер практически не ограничен.

Joomla – это система управления контентом, написанная на языке PHP и использующая в качестве хранилища содержания базу данных MySQL Joomla является свободным программным обеспечением, защищённым лицензией GPL. Одной из главных особенностей Joomla является относительная простота управления при практически безграничных возможностях и гибкости при изготовлении сайтов.

Joomla включает в себя различные инструменты для изготовления сайтов и web-сервисов. Важной особенностью системы является минимальный набор инструментов при начальной установке, который обогащается по мере необходимости. Это снижает загромождение административной панели ненужными элементами, а также снижает нагрузку на сервер и экономит пространство на хостинге [21].

Таким образом, существует множество средств разработки web-сервисов. У каждого средства разработки web-сервисов есть свои преимущества и недостатки. В данной работе для создания web-сервиса для оказания бытовых услуг была выбрана CMS платформа Joomla.

1.3 Формализованное описание технического задания

1. Общие сведения.

1.1. Название организации-заказчика.

Разработка web-сервис для оказания бытовых услуг «Everyday Services» выполнена в рамках выпускной квалификационной работы «Уральского государственного педагогического университета, института математики, физики, информатики и технологий, кафедры информационно-коммуникационных технологий в образовании».

1.2. Название продукта разработки (проектирования).

Web-сервис для оказания бытовых услуг «Everyday Services».

1.3. Назначение продукта.

Web-сервис для оказания бытовых услуг предназначен для предоставления информации по бытовым услугам и для взаимодействия клиентов: заказчиков и исполнителей.

Web-сервис должен содержать следующие функциональные возможности:

- регистрация пользователей;
- настройка учетных записей пользователя;
- просмотр списка услуг;
- выбор услуги;
- заказ услуги
- наличие среды для удаленного управления web-сервисом и его администрированием, и обслуживанием.
- Плановые сроки начала и окончания работ.

Начало работы: 01 июня 2017. Окончание работы: 10 декабря 2017.

2. Характеристика области применения продукта.

2.1. Процессы и структуры, в которых предполагается использование продукта разработки.

Web-сервис будет располагаться в общем доступе для всех пользователей сети Интернет.

2.2. Характеристика персонала (количество, квалификация, степень готовности).

Администратор – все заявки на утверждение услуг проходят через администратора данного web-сервиса, зарегистрировавшиеся новые пользователи должны пройти проверку администратора, чтобы новая учетная запись вступила в работу ее должен активировать администратор. Администратор следит за работой остального персонала.

Необходимы профессиональные навыки:

- Знание HTML, CSS, PHP, JavaScript и Adobe Flash;
- Умение составлять технические задания на разработку интернет-сервисов;
- Владение методами поисковой оптимизации сайта;
- Знание основ обеспечения комплексной безопасности веб-сайтов;
- Базовые знания PHP и MySQL;
- Система управления контентом, CMS.

Модераторы – модераторы занимаются оформлением web-сервиса и обновлением информации в web-сервисе. Количество модераторов – 2 человека.

Модератор должен обладать следующими качествами:

- Коммуникабельность. Модератору часто приходится общаться с пользователями сайта, писать им письма, разговаривать по телефону.
- Ответственность. Важно помнить, что от модератора зависит общая картина сайта, благодаря ему у пользователей складывается впечатление об интернет-ресурсе, поэтому степень ответственности за результаты работы у него должна быть высокой.
- Внимательность. Модератор не может пропустить ни одной ошибки в объявлении, материале или комментарии.

- Усидчивость. Необходимо иметь терпение, чтобы внимательно отсмотреть все материалы, сообщения и комментарии на сайте.
- Объективность. Модератор должен уметь отстраняться от своего личного отношения к теме обсуждения, быть объективным.

Маркетолог – маркетолог занимается рекламированием и продвижением web-сервиса. Количество маркетологов – 2 человека. Требуется высшее или средне-специальное образование маркетолога.

3. Требования к продукту разработки.

3.1. Требования к продукту в целом.

При реализации и использовании web-сервиса должны быть учтены требования к функциональным характеристикам, параметрам технических средств, информационной и программной совместимости.

3.2. Аппаратные требования.

Минимальные требования к программным и аппаратным средствам для нормального функционирования приложения:

Процессор: AMD или Intel с частотой 2,3 GHz и выше;

ОЗУ: 2 Gb и выше;

ОС: Windows 7 и выше;

Монитор: SVGA монитор;

Емкость ЖД: свободного места не менее 10 Gb;

Другие требования: сетевая карта, клавиатура, манипулятор мышь.

У каждого работника должно быть персональное цифровое устройство.

3.3. Указание системного программного обеспечения (операционные системы, браузеры, программные платформы и т.п.).

Программная система функционирует в среде Windows 7 и выше с любым браузером.

3.4. Указание программного обеспечения, используемого для реализации.

Denwer 3 PHP 5.3.13, MySQL 5.1, PostgreSQL 8.4 etc – локальный сервер;

Joomla – платформа для web разработки;

Google Chrome – браузер;

Photoshop CS6 – графический редактор.

3.5. Для сетевых систем – особенности реализации серверной и клиентской частей.

Серверная часть:

Web-сервис может быть расположен на хостинге с определенным доменом. Существуют разные тарифы на хостинг с разными ценами. Для того что бы web-сервис функционировал полноценно нужно выбрать для него домен, выбрать тариф для хостинга, зарегистрировать домен, оплатить заказ, получить доступ к выбранному домену. После чего можно переносить web-сервис на хостинг.

Клиентская часть:

Web-браузер является клиентской частью, с помощью браузера осуществляется доступ к web-сервису.

3.6. Форматы входных и выходных данных

В формате входных данных выступает:

- Со стороны заказчика – формирование заявки для заказа конкретной услуги.
- Со стороны исполнителя – формирование заявки на выполнение конкретной услуги.
- В формате выходных данных выступает:
- Со стороны заказчика – выполненная услуга.
- Со стороны исполнителя – получение оплаты за выполненную услугу.
- Источники данных и порядок их ввода в систему (программу), порядок вывода, хранения.

Источниками данных являются пользователи (клиенты). Пользователь регистрируется в системе, после чего получает доступ к выбору и заказу услуг.

Все заявки проходят через администратора web-сервиса. Все данные, вся информация хранятся на сервере.

3.7. Порядок взаимодействия с другими системами, возможности обмена информацией.

Взаимодействием с другими системами является только реклама web-сервиса.

3.8. Меры защиты информации.

Защита логина и пароля. Для того что бы войти в среду администрирования требуется ввести пароль от учетной записи администратора. Пароль администратора должен быть сложным, должен содержать в себе как минимум 8 символов букв и цифр.

Для большей безопасности можно скрыть вход в панель администратора, для этого нужно установить дополнительный плагин jSecure или его аналог kSecure.

Для сохранности информации о подключенных пользователях и их деятельности на web-сервисе необходимо периодически выполнять резервное копирование базы данных.

4. Требования к пользовательскому интерфейсу.

4.1. Общая характеристика пользовательского интерфейса.

В работе web-сервиса выделены три основные задачи: просмотр перечня услуг, заказ услуги, заказ услуги для ее выполнения. В соответствии с этим в интерфейсе предусмотрена панель с кнопками, с помощью которых можно перемещаться с одной страницы на другую.

4.2. Размещение информации на экране, дизайн экрана.

Дизайн web-сервиса приятный, навигация простая и удобная (рис. 2).

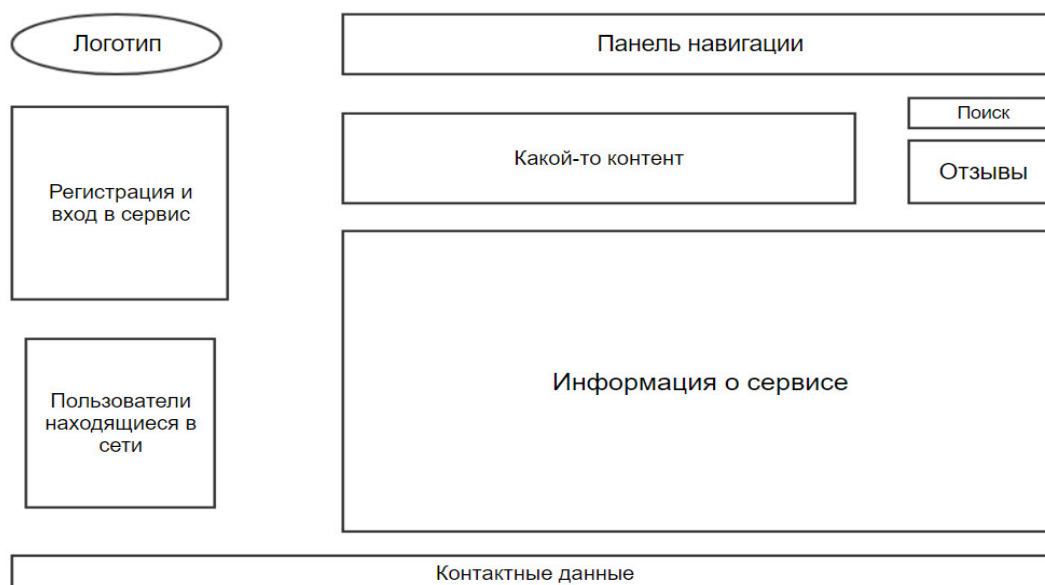


Рис. 2 Примерный дизайн экрана

4.3. Особенности ввода информации пользователем, представление выходных данных.

Для заказа услуги заказчику требуется пройти регистрацию и заполнить форму заявки:

- выбрать услугу из списка услуг;
- обозначить сумму оплаты за выполнение услуги;
- выбрать конкретный период для выполнения услуги;
- заполнить дополнительную информацию (адрес, доп. информацию об услуге, требования и т.п.), пройти простую проверку на анти-бот (например, $5+10=?$), нажать кнопку «Отправить».

Для выполнения услуги исполнителю пройти регистрацию, выбрать в форме из списка конкретную услугу и нажать кнопку «Отправить».

5. Требования к документированию.

5.1. Перечень сопроводительной документации.

Рекомендации по использованию среды администрирования Joomla.

Базовые рекомендации по безопасности Joomla.

5.2. Требования к содержанию отдельных документов.

5.3. Форматы документации.

Документация в формате PDF.

6. Порядок сдачи-приемки продукта.

В соответствии с графиком выполнения ВКР.

Глава 2. Практическая (оригинальная) часть

2.1 Модельные представления объекта разработки

Целью работы является создание web-сервиса для оказания бытовых услуг, следовательно, web-сервис будет создан основываясь на составленном алгоритме (рис. 3), в котором описывается, последовательность действий в работе web-сервиса.

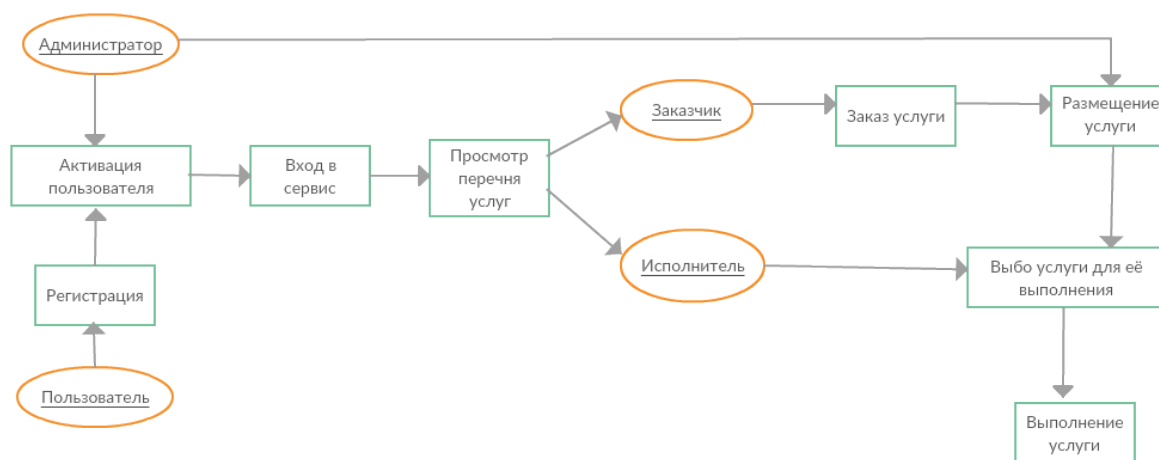


Рис. 3 Алгоритм работы web-сервиса

Пользователь имеет доступ через браузер к web-сервису, созданному на платформе CMS Joomla.

На выходе пользователь получает выполненную услугу либо определенную сумму денег за выполнение услуги, в зависимости от того, кто пользователь – заказчик или исполнитель.

Web-сервис, в свою очередь, состоит из страниц и расположенных на страницах модулей, компонентов и информации. Данный web-сервис имеет следующую структуру:

- **Главная страница** – на главной странице будет располагаться регистрация и вход в профиль учетной записи, сведения о данном web-сервисе, функция поиска, гостевая книга для пользовательских отзывов, контактные данные (Рис. 4);

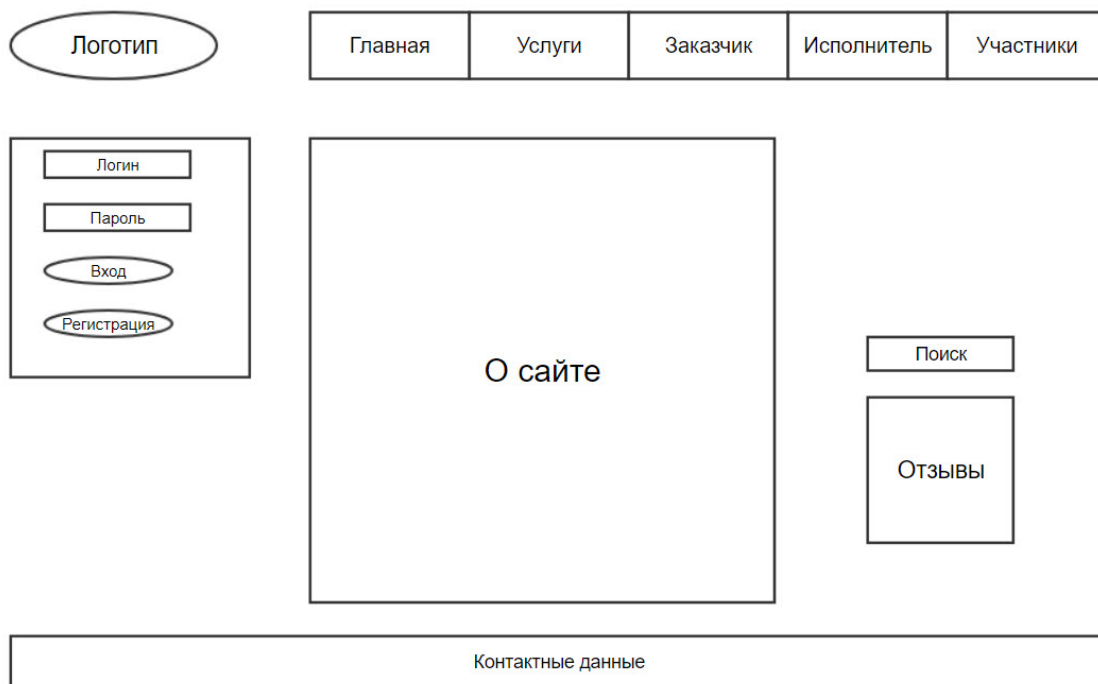


Рис. 4 Шаблон главной страницы

- **Услуги** – на этой странице будет представлен перечень услуг, которые смогут заказать или выполнить клиенты web-сервиса (Рис. 5);

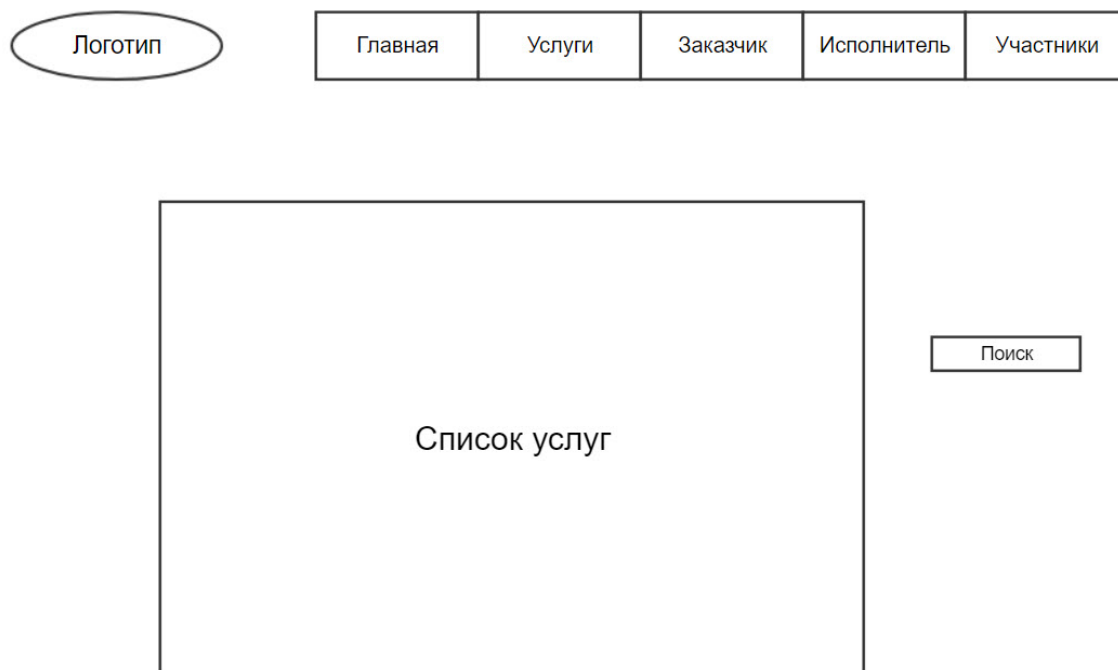


Рис. 5 Шаблон страницы - услуги

- **Заказчик** – на этой странице будет расположена форма заказа услуги (Рис. 6);

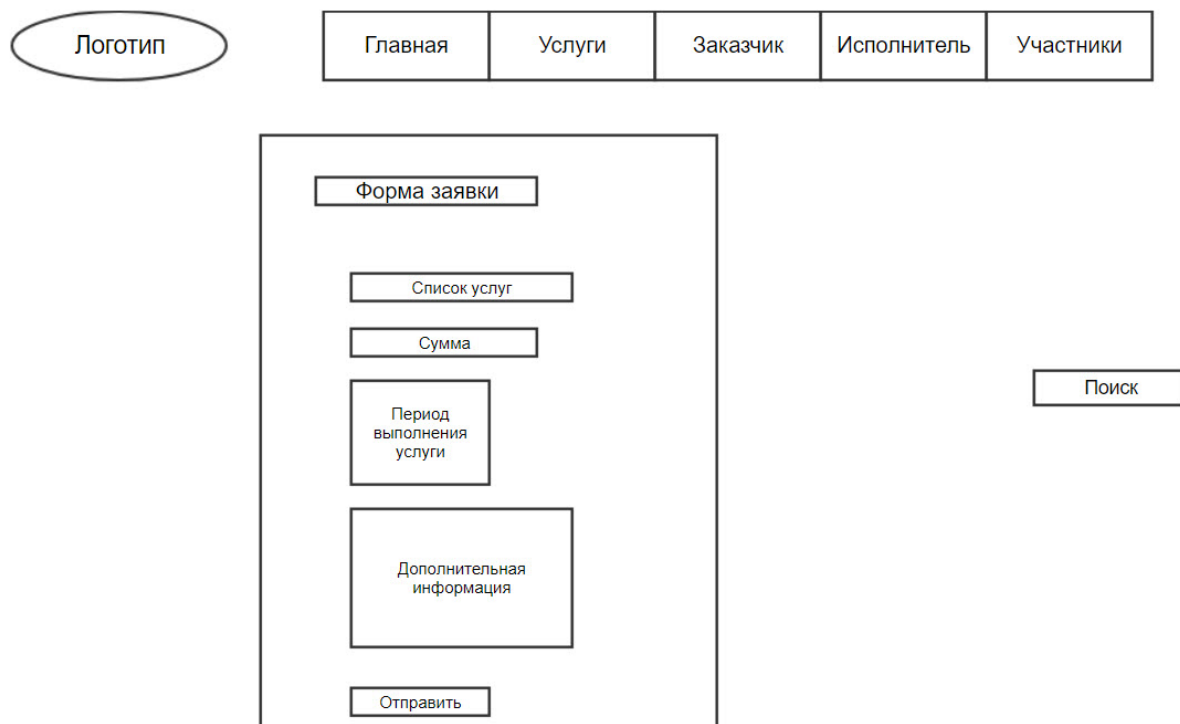


Рис. 6 Шаблон страницы - Заказчик

- **Исполнитель**— здесь будет расположена форма выбора услуги для ее выполнения (Рис. 7);



Рис. 7 Шаблон страницы - Исполнитель

- **Участники** – список пользователей, зарегистрированных в web-сервисе (Рис. 8);

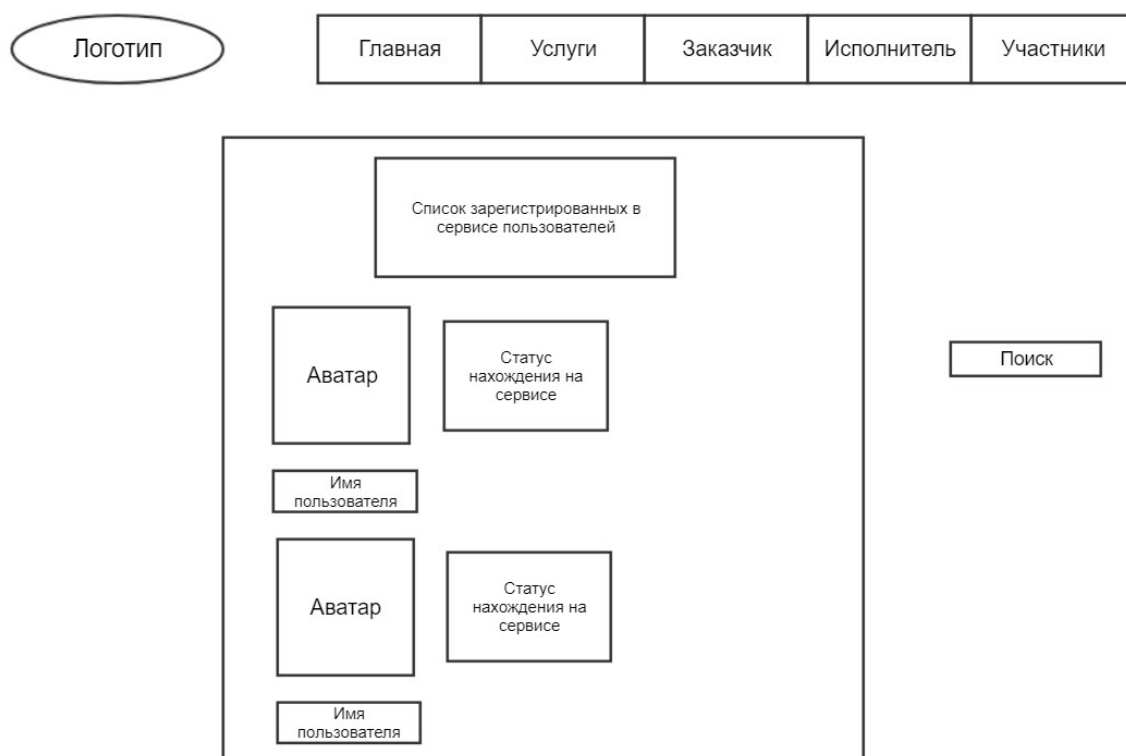


Рис. 8 Шаблон страницы - Участники

- **Мой Профиль** – в профиле будет отображаться информация о пользователе авторизовавшегося в web-сервисе (Рис. 9).

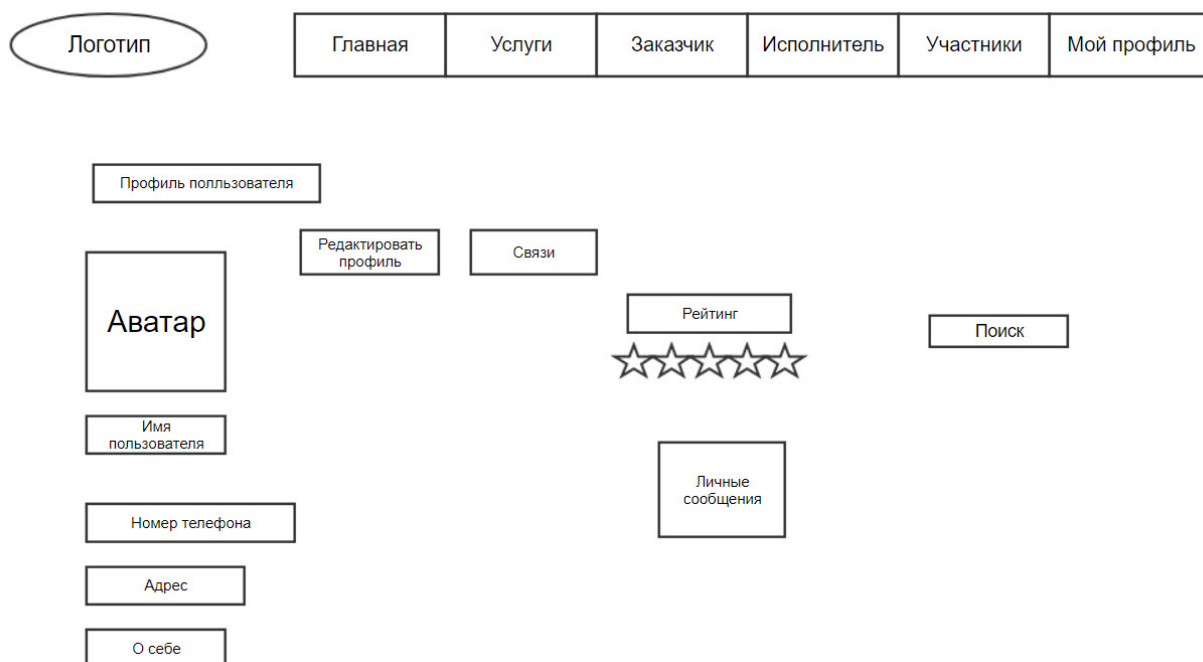


Рис. 9 Шаблон страницы – Мой профиль

2.2 Описание продукта (результата разработки)

Исходя из модельного представления и спроектированных шаблонов страниц для web-сервиса, был разработан web-сервис для оказания бытовых услуг.

Данный web-сервис представляет собой сервис, который дает возможность зарегистрированным в нем пользователям заказывать и выполнять бытовые услуги.

На главной странице web-сервиса представлена информация о web-сервисе, контактные данные, функция поиска, гостевая книга, панель авторизации. Для того что бы воспользоваться услугами сервиса пользователь должен авторизоваться в системе. (Рис. 10).

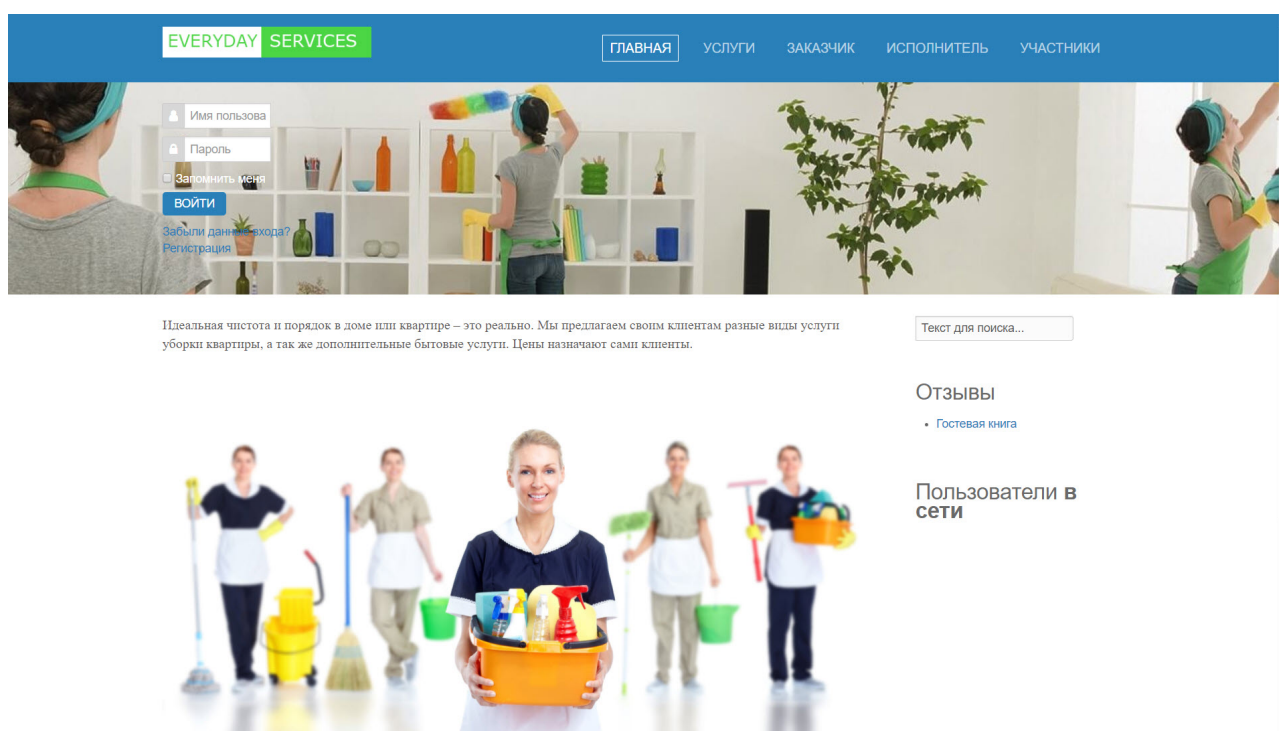


Рис. 10 Главная страница

При первом входе в web-сервис пользователь должен выполнить регистрацию (Рис. 11). Поля, обозначенные звездочкой обязательны для заполнения. Дополнить информацию о себе пользователь сможет, зайдя в свой профиль, после того как Администратор подтвердит заявку на регистрацию.

Рис. 11 Регистрация в системе web-сервиса

Администратор управляет списком пользователей через административную панель при помощи модуля Community Builder. Администратор может блокировать, активировать, создавать, удалять пользователей, редактировать и просматривать личную информацию в профиле пользователя (Рис. 12).

#	Имя	Имя пользователя	На сайте	Группа	Адрес электронной почты	Регистрация	Последний визит на сайт	Заблокирован	Включен	Подтвержден	Одобен	ID
1	Татьяна Исакова	admin	✓	Super Users	imbershka@gmail.com	09/24/2017 12:04:58	11/30/2017 12:51:58	✗	✓	✓	✓	371
2	Иван Иванов	User1	✓	Registered	user@mail.ru	09/28/2017 10:50:02	11/30/2017 12:40:07	✗	✓	✓	✓	372
3	Михаил	User2	✗	Registered	user2@mail.ru	11/20/2017 06:39:50	11/22/2017 15:48:32	✗	✓	✓	✓	373
4	Модератор	Moderator	✗	Manager	moder@gmail.com	11/21/2017 06:48:32		✗	✓	✓	✓	374
5	User3	User3	✗	Registered	user3@mail.ru	11/22/2017 14:51:16	11/22/2017 15:49:04	✗	✓	✓	✓	375
6	User4	User4	✗	Registered	user4@mail.ru	11/23/2017 13:17:52	11/23/2017 13:23:57	✗	✓	✓	✓	376
7	Модератор2	Moderator2	✗	Manager	moder2@mail.ru	11/30/2017 12:56:48		✗	✓	✓	✓	377
8	Маркетолог	marketer	✗	Super Users	marketer@mail.ru	11/30/2017 13:01:13		✗	✓	✓	✓	378
9	Маркетолог2	marketer2	✗	Super Users	marketer2@mail.ru	11/30/2017 13:02:40		✗	✓	✓	✓	379

Рис. 12 Управление пользователями через админ-панель

После того как заявка на регистрацию одобрена Администратором, на указанный пользователем электронный адрес приходит сообщение с действующим логином и паролем. После авторизации в web-сервисе пользователь может заказывать услуги. Ознакомиться со списком услуг, которые предлагает сервис, пользователь может на странице «Услуги» (Рис. 13, 14).

Планируете сделать уборку, но ленитесь? Готовы заплатить за работу? Её с удовольствием сделают за вас!
С EverydayServices вы сэкономите кучу времени и сил, заказывая услугу и назначая самостоятельно цену для нее.
Если у вас есть время и силы и вы готовы заработать, вам осталось только выбрать подходящую для вас работу и выполнить ее. Выбор за вами.

Пользователи в сети

User1

Перечень услуг:

[Генеральная уборка квартиры](#)

[Уборка лоджий, балконов](#)

[Уборка кухни](#)

[Мытье полов](#)

[Мытье окон](#)

[Уборка после ремонта](#)



[Заказать услугу](#)

[Выполнить услугу](#)

Рис. 13 Перечень услуг

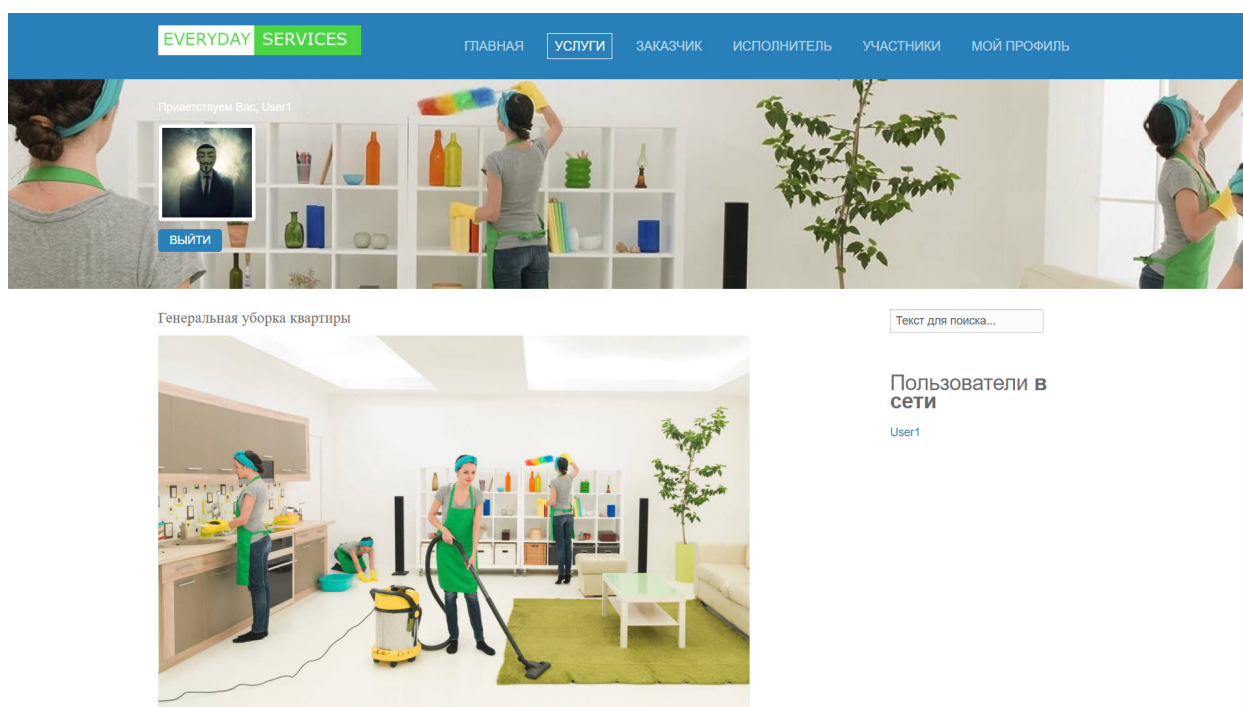


Рис. 14 Услуга – генеральная уборка квартиры

Для того что бы заказать определенную услугу пользователь должен зайти на вкладку «Заказчик», заполнить форму заявки для заказа услуги. Выбрать определенную услугу из списка, обозначить цену, указать период выполнения услуги, указать имя, фамилию, номер телефона, электронный адрес, свой адрес (на случай если адрес данные указанные в профиле на данный момент не являются действительными), дополнительную информацию (если требуется) (Рис. 15, 16).

 The screenshot shows the 'Оформить заказ' (Place order) form. The form is divided into several sections:
 1. 'Выберите услугу:' (Select service): A dropdown menu with 'Генеральная уборка квартиры' (General apartment cleaning) selected.
 2. 'Оплата:' (Payment): Fields for 'Выплата:' (Payment) with a unit 'руб.' (rub.) and 'Сумма:' (Amount).
 3. 'Выберите период для выполнения услуги:' (Select period for service execution): Fields for 'Дата начала выполнения услуги:' (Start date of service execution) and 'Дата окончания выполнения услуги:' (End date of service execution), both set to 'Ноябрь-30-2017'.
 4. 'Личная информация:' (Personal information): Fields for 'Имя:' (Name) and 'Фамилия:' (Surname).
 On the right side of the form, there is a search bar with the placeholder 'Текст для поиска...' and a section titled 'Пользователи в сети' (Users online) showing 'User1' as online.

Рис. 15 Заказ услуги, часть 1



Рис. 16 Заказ услуги, часть 2

После того как пользователь оформил заявку и нажал на кнопку отправить, заявка приходит на электронный адрес Администратора. Администратор через модуль «QuickForm» указывает название услуги, цену за выполнение услуги, которую обозначил заказчик и добавляет услугу на web-сервис (Рис. 17).

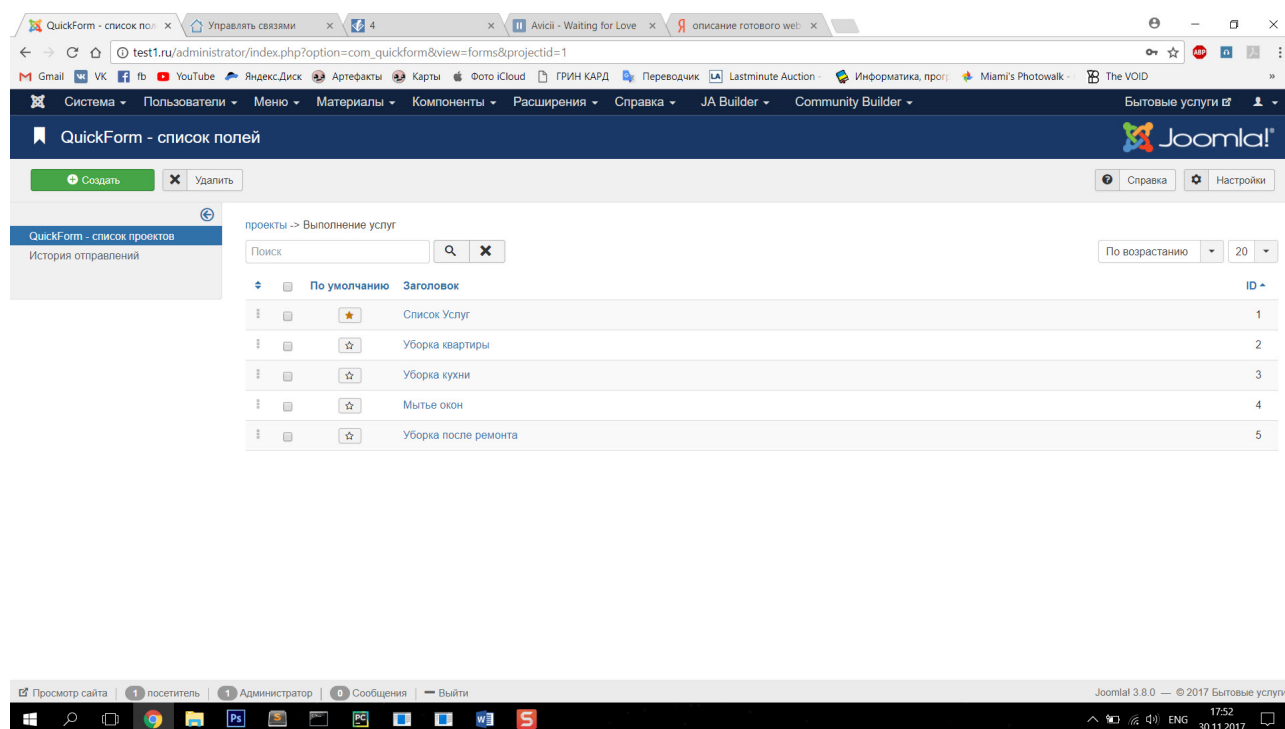


Рис. 17 Редактирование списка услуг для выполнения

Добавленная заявка появляется на странице «Исполнитель» в форме для выполнения услуги (Рис. 18), в этой форме представлен список актуальных заявок с определенной стоимостью. Исполнитель выбирает услугу, которая ему подходит, нажимает кнопку отправить (Рис. 19). Заявка приходит на электронный адрес Администратора. Администратор связывает двух пользователей заказчика и исполнителя, предоставляя заказчику данные исполнителя, исполнителю данные заказчика.

Так как определяется период времени для выполнения услуги, список услуг на странице «Исполнитель» своевременно редактируется Администратором web-сервиса.

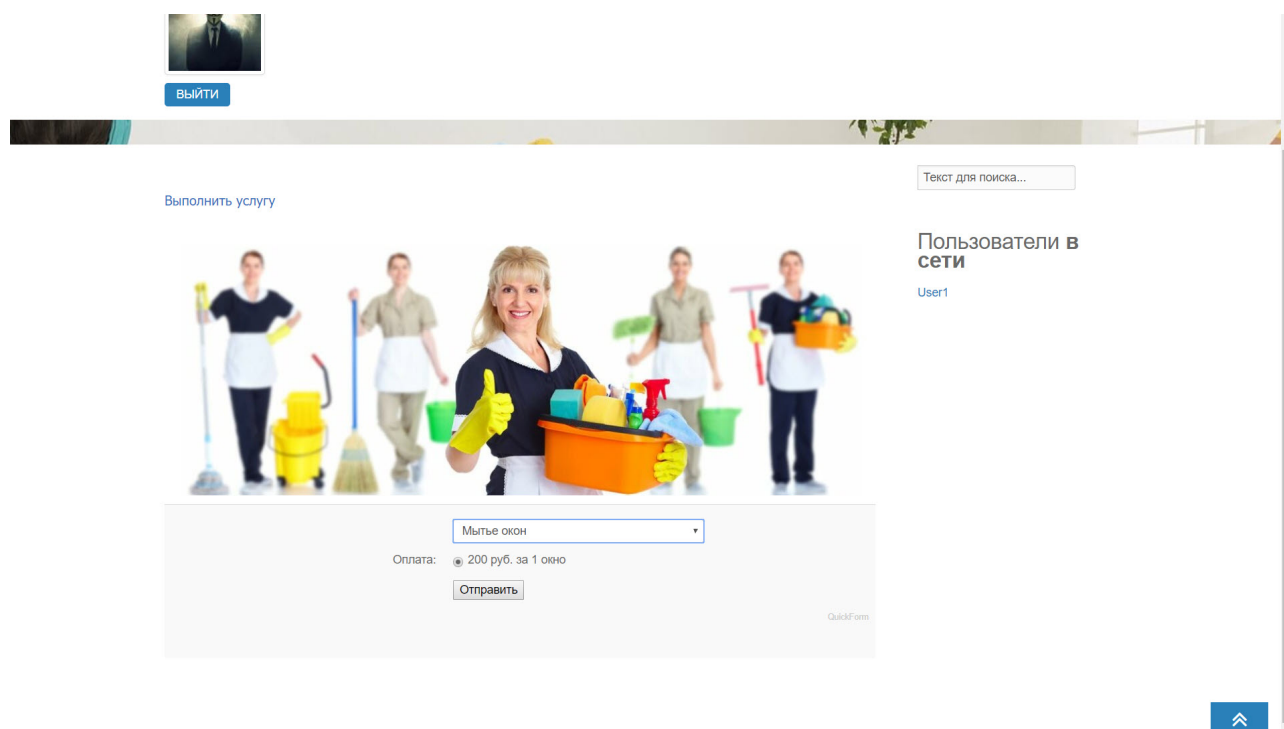


Рис. 18 Форма выбора услуги для её выполнения

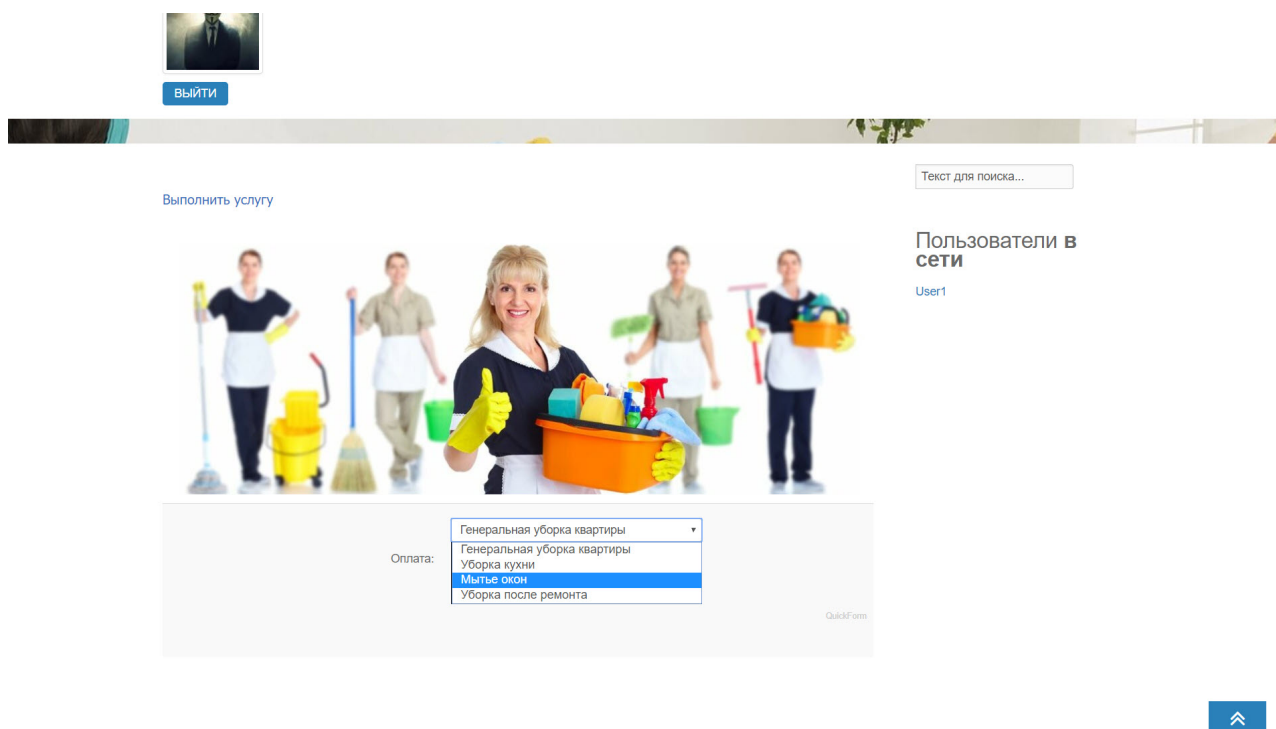


Рис. 19 Выбор услуги для её выполнения

В web-сервисе имеется возможность просмотра списка участников сервиса на вкладке «Участники» (Рис. 20), а также просмотр профиля пользователей, добавление пользователей в список друзей, возможность отправки личного сообщения. Пользователи могут ставить друг другу оценки, тем самым повышая или понижая рейтинг пользователя (Рис. 21).

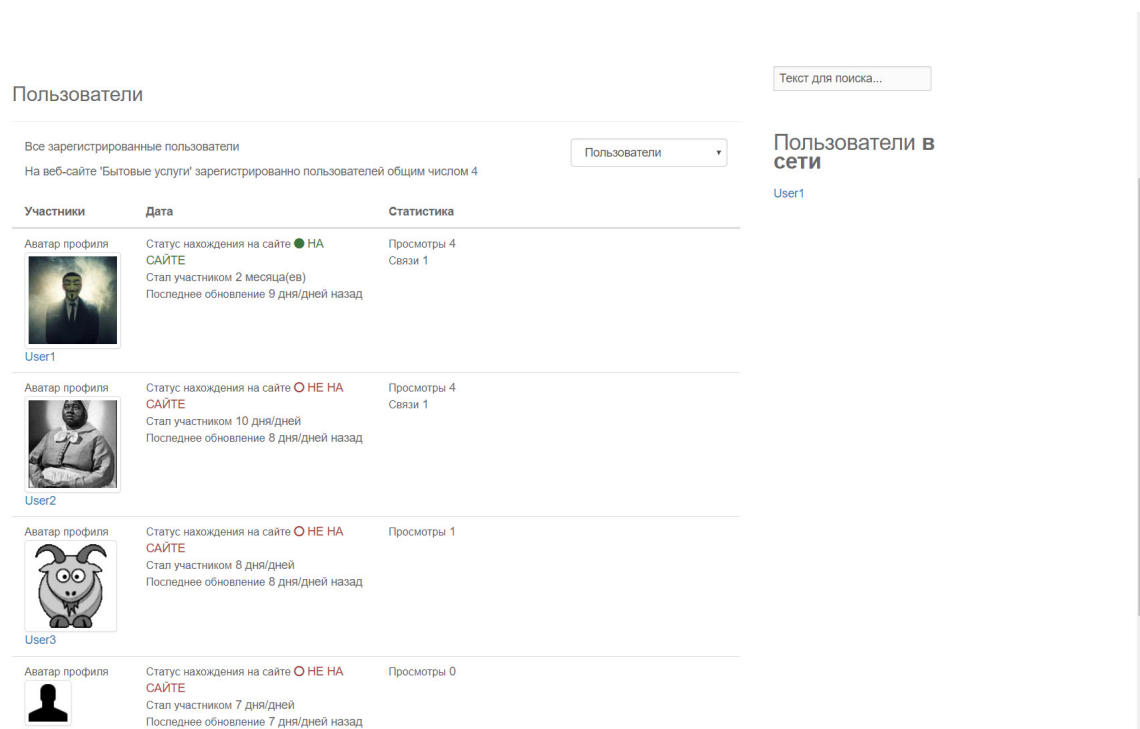


Рис. 20 Список участников web-сервиса

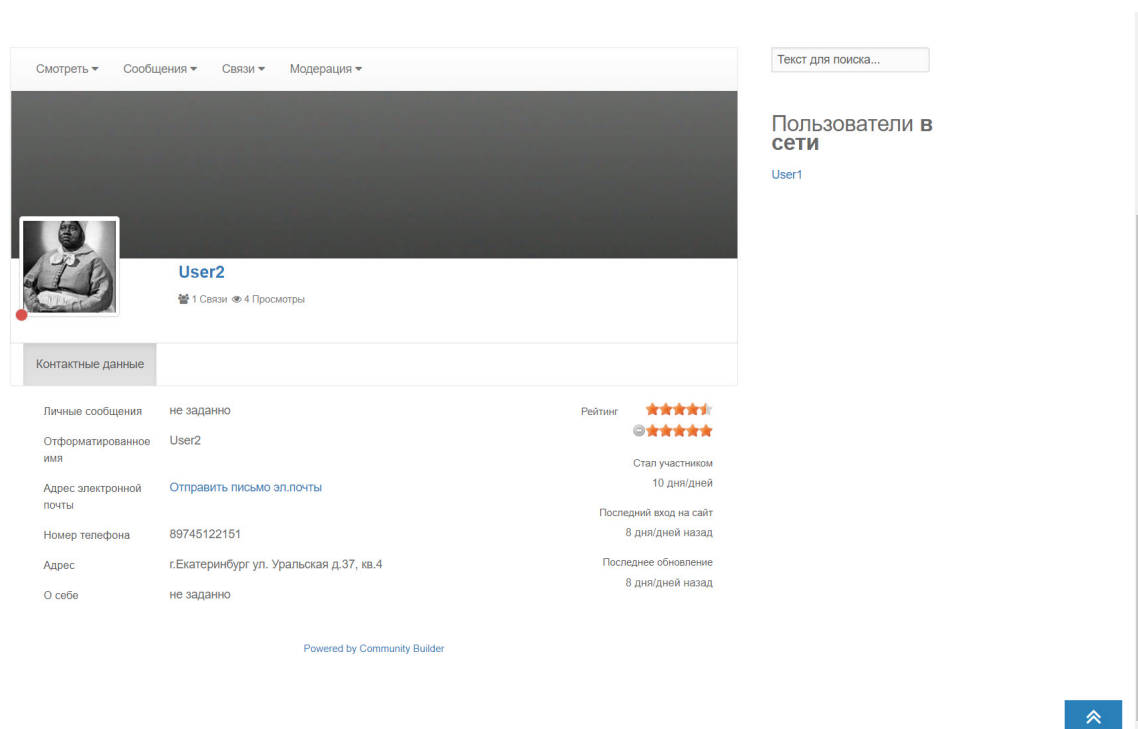


Рис. 21 Просмотр профиля пользователя

Пользователи могут оставлять свои отзывы в системе «Гостевая книга», которая расположена на главной странице сервиса. Комментарии просматривают Модераторы и вносят соответствующие изменения в web-сервисе. Также Модераторы следят за функционалом, внешним видом и работой web-сервиса.

Рекламой и раскруткой web-сервиса занимаются Маркетологи. У всего персонала имеется доступ к web-сервису и определенные права для работы с web-сервисом.

2.3 Результаты апробации, техническая документация

Для данного web-сервиса прилагается инструкция для работы с ним:

1. Инструкция для работы с web-сервисом в формате PDF.
2. Видео-обзор web-сервиса.

3. С помощью сервиса «Google формы» был составлен опрос, к опросу прилагается видео-обзор web-сервиса (Приложение 2.). Для прохождения опроса, отвечающим нужно было перейти по [ссылке](#) и ответить на вопросы. В результате опроса двадцати человек, были получены результаты (Рис. 25, 26, 27, 28, 29).

1) Как вы оцениваете внешний вид web-сервиса?

20 ответов

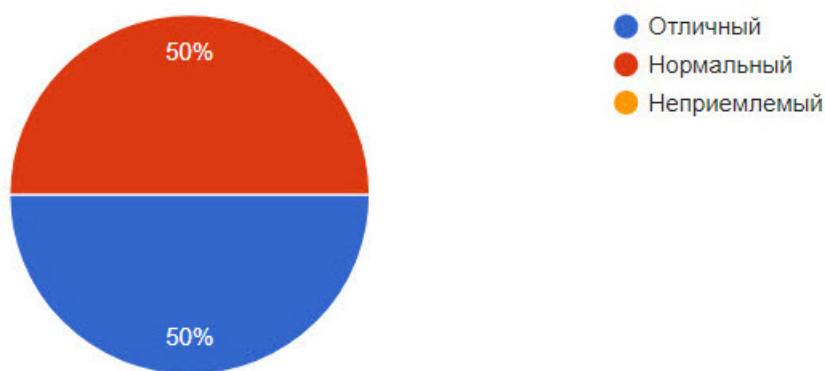


Рис. 25 Статистика (1 вопрос)

В результате проведения опроса, мы получили следующие результаты:

Внешний вид сервиса первая половина отвечающих оценила, как отличный, вторая половина отвечающих считает, что внешний вид сервиса нормальный.

Работу web-сервиса 9 отвечающих оценили на 10 баллов, 6 отвечающих на 9 баллов, 4 отвечающих на 8 баллов и один отвечающий на 7 баллов.

40% отвечающих предпочли бы использовать на сервисе заказ и выполнение услуг, 20% выполнение услуг и 40% заказ услуг.

12 отвечающих считают, что в сервисе достаточно услуг, которые предложены на данный момент и дополнять их не нужно. И всего два отвечающего считают, что список услуг стоит дополнить.

Двое отвечающих считают, что в сервисе не хватает функционала (опций, функций), остальных отвечающих все устраивает.

15% отвечающих являются специалистами IT-технологий, 25% отвечающих интересуются IT-технологиями и 60% никак не связаны с IT.

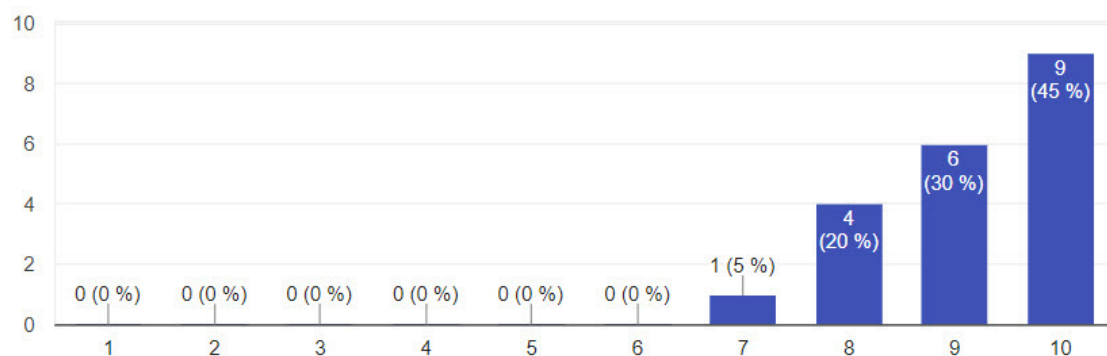
85% отвечающих никогда не использовали подобный сервис, 15% ранее пользовались подобным сервисом.

65% отвечающих хотят в дальнейшем пользоваться услугами данного web-сервиса, 35% не уверены в том, что будут использовать этот сервис.

Стоит отметить, что ни одного плохого отзыва о сервисе на данный момент не выявилось.

2) Как вы оцениваете работу web-сервиса по шкале от 1 до 10?

20 ответов



3) Какой вид сервиса вы предпочли бы использовать?

20 ответов

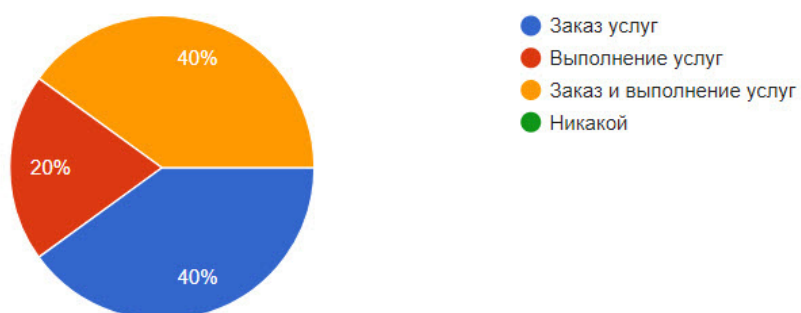


Рис. 26 Статистика (2-3 вопрос)

4) Стоит ли дополнить перечень услуг на нашем web-сервисе? Если да, то перечислите какие услуги стоит добавить в перечень. Если нет, напишите почему не стоит.

14 ответов

нет (4)
Нет (3)
да (2)
Нет, все нравится
Нет, все здорово сделано
Достаточно услуг
Нет. Все в достатке
достаточно услуг

Рис. 27 Статистика (4 вопрос)

5) Стоит ли добавить в сервис дополнительные опции и функции, если да то какие?

13 ответов

нет (6)
Нет (5)
какие-нибудь еще
да

6) Являетесь ли вы специалистом в сфере IT-технологий?

20 ответов

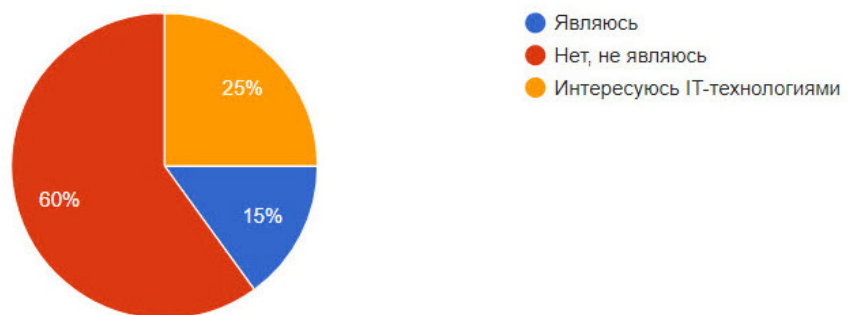
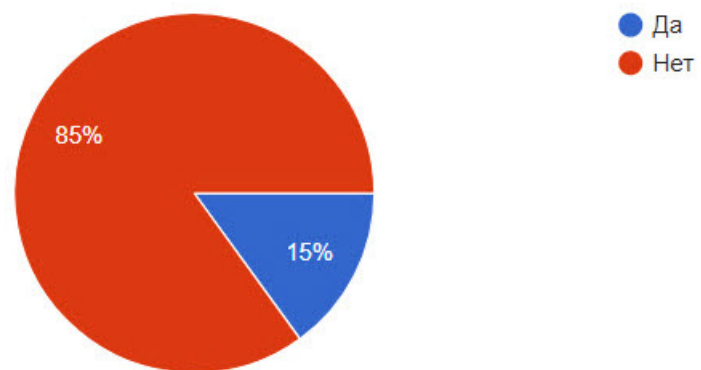


Рис. 28 Статистика (5-6 вопрос)

7) Использовали ли вы подобный web-сервис ранее?

20 ответов



8) Будете ли вы использовать данный сервис?

20 ответов

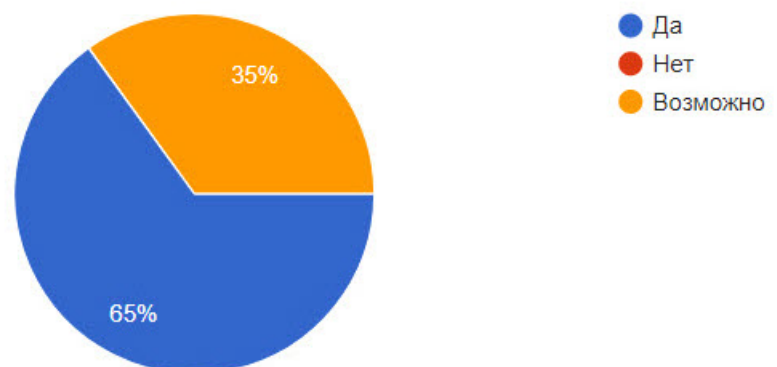


Рис. 29 Статистика (7-8 вопрос)

Заключение

Использование услуг web-сервиса является одной из наиболее перспективных форм ведения бизнеса и организации работы в современном обществе. Использование web-сервиса для оказания бытовых услуг решает проблему поиска исполнителей для выполнения бытовых услуг и позволит заработать клиентам деньги за выполнение бытовых работ.

В данной работе была поставлена следующая цель: разработать web-сервис для оказания бытовых услуг.

В процессе достижения поставленной цели были выполнены следующие задачи:

1. Изучены основные понятия разработки web-сервисов;
2. Рассмотрены самые популярные и востребованные CMS платформы и языки для разработки web-сервисов;
3. Изучена выбранная система для разработки web-сервиса для оказания бытовых услуг;
4. Разработан web-сервис для оказания бытовых услуг;
5. Разработана инструкция по использованию web-сервиса для пользователя и администратора.

Таким образом, следует считать, что результаты разработки соответствуют всем требованиям технического задания, поставленная цель достигнута. Работа носит законченный характер.

Список информационных источников

1. Encyclopedia of virtual communities and technologies. Subhasish Dasgupta, editor. Idea Group Reference, USA, 2006.
2. Hypertext Transfer Protocol --HTTP/1.1 <http://www.ietf.org/rfc/rfc2616.txt> (дата обращения 11.10.2017).
3. Mel P., Grance T. The NIST Definition of Cloud Computing. Recommendations of the National Institute of Standards and Technology <http://csrc.nist.gov/publications/PubsSPs.html#800-145> (дата обращения 17.10.2017).
4. SOAP Version 1.2 Part 0: Primer (Second Edition) <http://www.w3.org/TR/2007/REC-soap12-part0-20070427/> (дата обращения 17.10.2017).
5. SOAP Version 1.2 Part 1: Messaging Framework (Second Edition) <http://www.w3.org/TR/2007/REC-soap12-part1-20070427/> (дата обращения 17.10.2017).
6. SOAP Version 1.2 Part 2: Adjuncts (Second Edition) <http://www.w3.org/TR/2007/REC-soap12-part2-20070427/> (дата обращения 17.10.2017).
7. Web-сервис Энциклопедический словарь СМИ. <https://smi.academic.ru/7/WEB-сервис> (дата обращения: 01.11.2017).
8. Wikipedia <https://ru.wikipedia.org/wiki/Joomla!> (дата обращения 18.10.2017).
9. Wix конструктор сайта <https://ru.wix.com/> (дата обращения 18.10.2017).
10. А.В Сычев Web-технологии <http://www.intuit.ru/department/internet/webtechno/> (дата обращения 17.10.2017).
11. А.В. Сычев Перспективные технологии и языки веб-разработки <http://www.intuit.ru/department/internet/atlwebdev/> (дата обращения 17.10.2017)

12. Академик Википедия <https://dic.academic.ru/dic.nsf/ruwiki/1465> (дата обращения 18.10.2017).
13. Басс Л., Клементс П., Кацман Р. Архитектура программного обеспечения на практике. СПб.: Питер, 2006 – 576 с.
14. Блог Vaden Pro <https://vaden-pro.ru/blog/drupal/что-такое-drupal> (дата обращения 18.10.2017).
15. Викиреальность статья Веб-сервис <http://www.wikireality.ru/wiki/Веб-сервис> (дата обращения 09.10.2017).
16. Г.Л. Багиев, И. В. Успенский, В.И. Ченцов Интерактивные модели маркетинговых решений на виртуальных рынках. СПб: СПбГУЭиФ, 2008. 152 с.
17. Герберт Шилдт. Java 8. Полное руководство, 9-е издание = Java 8. The Complete Reference, 9th Edition. — М.: «Вильямс», 2015 (дата обращения 18.10.2017).
18. Как создать сайт на WordPress: полное руководство для новичков <https://texterra.ru/blog/kak-sozdat-sayt-na-wordpress-polnoe-rukovodstvo-dlya-novichkov.html> (дата обращения 18.10.2017).
19. Настройка и конфигурация HTTP сервера Apache на Windows <http://zametkinapolyah.ru/servera-i-protokoly/что-такое-http-server-apache-ustanovka-veb-servera-apache-2-4-na-windows-gde-skachat-apache-2-4-dlya-windows.html> (дата обращения 18.10.2017).
20. Первые шаги руководство Joomla <http://joomla.ru/docs/administrator/joomla3-start/1742-что-такое-> (дата обращения 18.10.2017).
21. Программные средства для разработки web-сайта <http://zdamsam.ru/a19366.html> (дата обращения 18.10.2017).
22. Система управления содержимым https://ru.wikipedia.org/wiki/Система_управления_содержимым (дата обращения 12.10.2017).

23. Рябов В.А., Несвижский А.И. Современные web-технологии. М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2006 – 432 с.
24. С. Лукошкин Блог Что такое CMS (движок) и почему это важно для бизнес-сайта? <http://partizzan.ru/chto-takoe-cms-dvizhok-i-pochemu-eto-vazhno-dlya-biznes-sajta/> (дата обращения 12.10.2017).
25. С.А. Колюшин Принципы построения веб-сайтов с использованием CMS http://nic1.ifmo.ru/publications/articles/MajorRead2009/MREAD2009_KolushinCA.pdf (дата обращения 11.10.2017).
26. С.В. Одиночкина Учебное пособие Разработка WEB-сервисов в инфокоммуникационных системах 2013. 124 с.
27. Сиддикуи Б. Безопасность Web-сервисов <http://www.iso.ru/rus/document5972.phtml> (дата обращения 14.10.2017).
28. Словарь компьютерных терминов <http://computer.slovaronline.com/B/761-VEB-SERVIS> (дата обращения 17.10.2017).
29. Сорокин, А.В. Проблемы интернета: Взаимодействие Web-сервисов -новый этап развития интернета Электронный ресурс http://www.aselibrary.ru/datadocs/doc_754ge.pdf (дата обращения 01.11.2017).
30. Средства создания Web-сайтов. Введение в разработку Web-приложений Лекция <http://5fan.ru/wievjob.php?id=51474> (дата обращения 17.10.2017).
31. Статьи о программировании <https://bunkerbook.ru/articls/129-samye-populyarnye-i-vostrebovannye-yazyki-programmirovaniya-v-2017-godu-html/> (дата обращения 18.10.2017).
32. Столлигнс В. Компьютерные сети, протоколы и технологии Интернета. BHV, 2005 – 817 с.

33. Студенческая библиотека онлайн Content management system (CMS) система wordpress

http://studbooks.net/1160800/informatika/takoe_sistema_upravleniya_kontentom
(дата обращения 11.10.2017).

34. Файловый архив студентов. URL: <https://studfiles.net/preview/1772796/>
(дата обращения 10.10.2017).

35. Фастовский Э.Г. Сервис-ориентированные технологии интеграции информации. Лекции. Харьков: НТУХПИ, 2011 – 50 с
<http://www.twirpx.com/file/1112914/> (дата обращения 01.10.2013).

36. Что такое CMS или система управления контентом
<http://blogwork.ru/chto-takoe-cms-ili-sistema-upravleniya-kontentom/> (дата обращения 11.10.2017).

37. Электронная библиотека студента. Библиофонд. Тема: Web–сервисы
<http://bibliofond.ru/view.aspx?id=34008> (дата обращения 17.10.2017).

Приложения

Приложение 1.

Инструкция для работы с web-сервисом

Инструкция для пользователя

При первом входе в web-сервис выполните регистрацию. Заполните форму регистрации. Ячейки, отмеченные звездочкой, являются обязательными для заполнения (Рис. 1).

Рис. 1 Регистрация в web-сервисе

Далее проверьте электронную почту, которую указали при регистрации. На ваш почтовый ящик придет письмо от Администратора с подтверждением «о регистрации учетной записи», в котором будет указан ваш действующий логин и пароль.

Введите логин и пароль в форму входа в web-сервис. Появится сообщение с приветствием, теперь вы можете пользоваться услугами нашего сервиса.

Для того что бы редактировать информацию в своем профиле зайдите на вкладку «Мой профиль». Здесь вы можете добавить, исправить свою личную информацию, добавить фотографию, изменить внешний вид профиля. Управлять списком друзей (Рис. 2).

Редактировать личные данные

Контактные данные

Полное Портрет

Имя пользователя

User1

Адрес электронной почты

user@mail.ru

Пароль

Подтвердите пароль

Текущий Пароль

Язык сайта

- Использовать по умолчанию -

Редактор

- Использовать по умолчанию -

Часовой пояс

- Использовать по умолчанию -

Номер телефона

89574123697

Адрес

г.Екатеринбург, ул.Мира д.7, к.47

О себе

Заказчик

ОБНОВИТЬ

ОТМЕНИТЬ

Powered by Community Builder

Текст для поиска...

Пользователи в сети

User1

Рис. 2 Редактирование профиля пользователя

Для знакомства с услугами web-сервиса, перейдите на вкладку «Услуги», на этой странице предоставлен перечень услуг, которые оказывает web-сервис (Рис. 3).

Планируете сделать уборку, но ленитесь? Готовы заплатить за работу? Ее с удовольствием сделают за вас! С EverydayService вы сэкономите кучу времени и сил, заказывая услугу и платя за нее самостоятельно. Если у вас есть время и силы и вы готовы заработать, вам осталось только выбрать подходящую для вас работу и выполнить ее. Выбор за вами.

Перечень услуг:
Генеральная уборка квартиры
Уборка лоджий, балконов
Уборка кухни
Мытье полов
Мытье окон
Уборка после ремонта



Заказать услугу

Выполнить услугу

Текст для поиска...

Пользователи в сети

User1

Рис. 3 Перечень услуг

Для того чтобы заказать услугу перейдите на вкладку «Заказчик» (Рис. 4). Заполните форму заказа услуги:

Выберите интересующую вас услугу из списка услуг;

Обозначьте цену за выбранную услугу;

Выберите период для выполнения услуги;

Укажите имя, фамилию, номер телефона, адрес электронной почты и дополнительную информацию по услуге;

Нажмите кнопку «Отправить».

Заявка будет отправлена на электронный ящик Администратора. Ждите, когда с вами свяжутся.

The screenshot shows a web form titled 'Оформить заказ' (Place order). It includes a search bar at the top right labeled 'Текст для поиска...'. Below the title, there are several sections: 'Выберите услугу' (Select service) with a dropdown menu showing 'Генеральная уборка квартиры'; 'Оплата' (Payment) with fields for 'Валюта:' (Currency) set to 'руб.' and 'Сумма:' (Amount); 'Выборите период для выполнения услуги' (Select period for service execution) with a date field set to 'Ноябрь-2017'; 'Выборите конечный срок выполнения услуги' (Select final deadline for service execution) with a date field set to 'Ноябрь-2017'; and 'Личная информация' (Personal information) with fields for 'Имя:' (Name) and 'Фамилия:' (Surname). On the right side, there is a section 'Пользователи в сети' (Users online) showing 'User1'.

Рис. 4 Заказ услуги

Для того что бы выполнить услугу перейдите на вкладку «Исполнитель» (Рис. 5). Выберите из списка услугу, которую желаете выполнить. Нажмите кнопку «Отправить».

Заявка будет отправлена на электронный ящик Администратора. Ждите, когда с вами свяжутся.

The screenshot shows a web page titled 'Выполнить услугу' (Perform service). At the top, there is a user profile picture and a 'Выйти' (Logout) button. Below the title, there is a large banner image showing several people in cleaning uniforms. Below the banner, there is a dropdown menu for selecting a service, currently showing 'Мытье окон' (Window cleaning). Below this, there is a price field 'Оплата:' (Payment) showing '200 руб. за 1 окно' (200 rub. for 1 window). At the bottom, there is a button labeled 'Отправить' (Send). On the right side, there is a section 'Пользователи в сети' (Users online) showing 'User1'.

Рис. 5 Выбор услуги для выполнения

Для того что бы просмотреть профиль интересующего вас пользователя перейдите на вкладку «Участники» (Рис. 6).

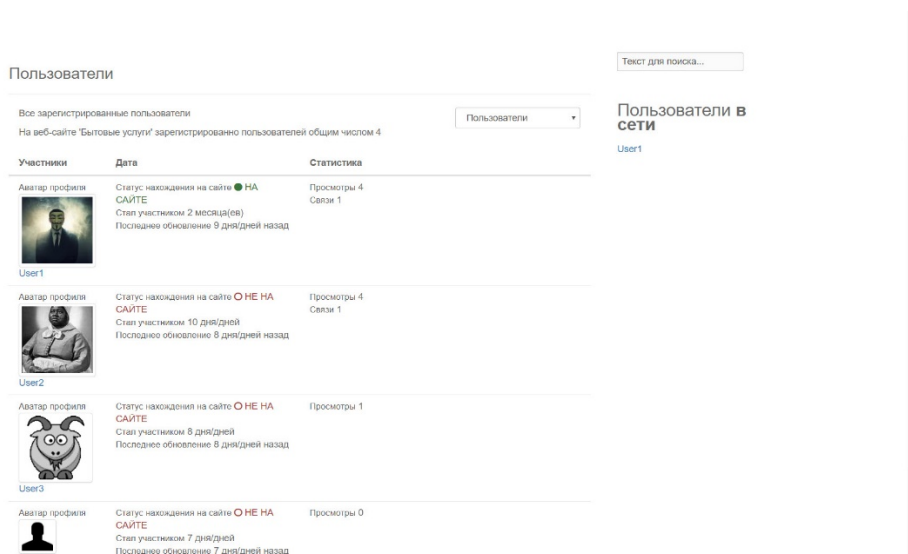


Рис. 6 Список участников

В списке пользователей выберите пользователя перейдите в его профиль. Вы можете повысить или понизить рейтинг пользователя, для этого поставьте оценку пользователю в категории «Рейтинг». Также можете отправить личное сообщение пользователю, добавить или удалить из списка друзей, пожаловаться на пользователя (Рис. 7).

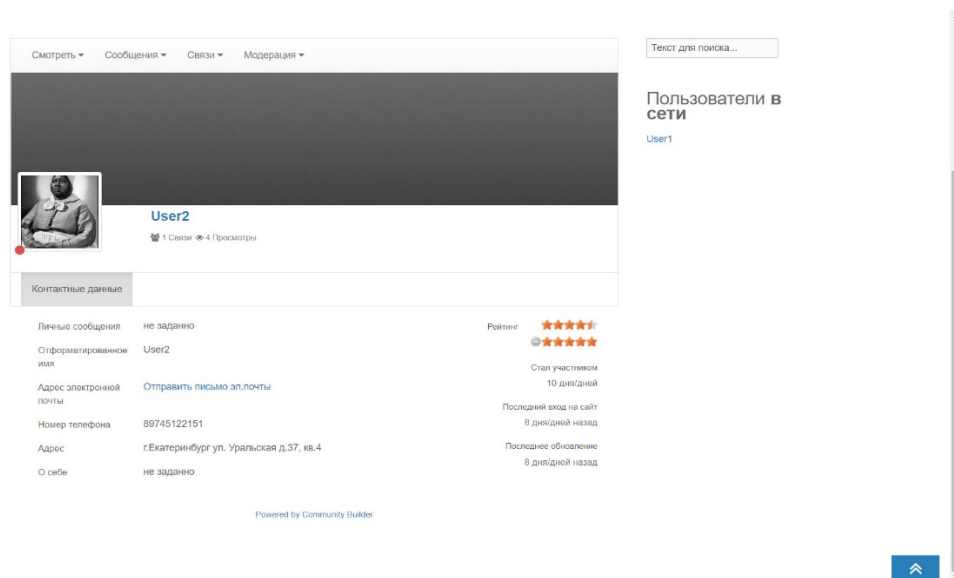


Рис. 7 Просмотр профиля пользователя

Для того что бы оставить отзыв о web-сервисе перейдите на главную страницу, вкладка «Главная», откройте гостевую книгу в категории «Отзывы» (Рис. 8).

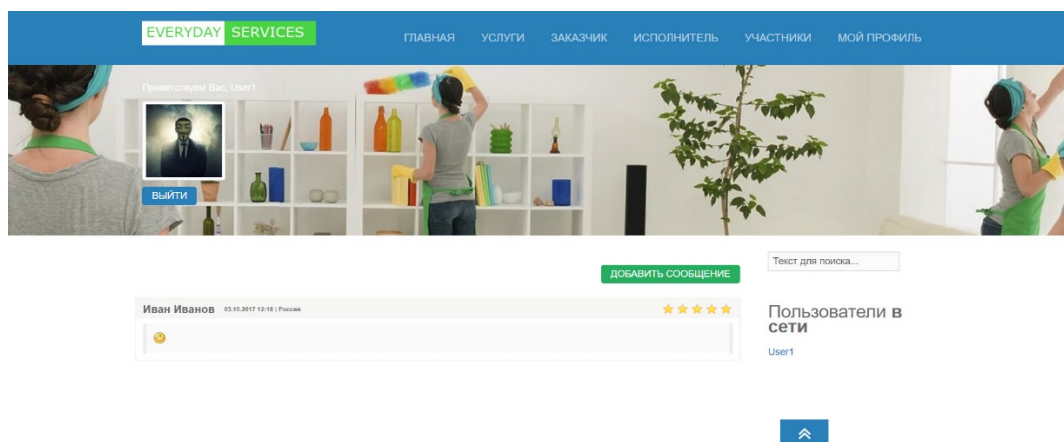


Рис. 8 Отзывы

Данные обратной связи указаны на главной странице web-сервиса в левом нижнем углу.

Инструкция для Администратора

Главная задача Администратора состоит в том, что он должен оперативно реагировать на заявки от пользователей, своевременно добавлять и редактировать эти заявки в web-сервисе. Связывать между собой заказчиков и исполнителей. Одобрять заявки от пользователей на регистрацию в web-сервисе.

Для входа в административную панель введите в адресной строке, после адреса сервиса administrator. В появившемся окне авторизации введите действующий логин и пароль Администратора (Рис. 9).

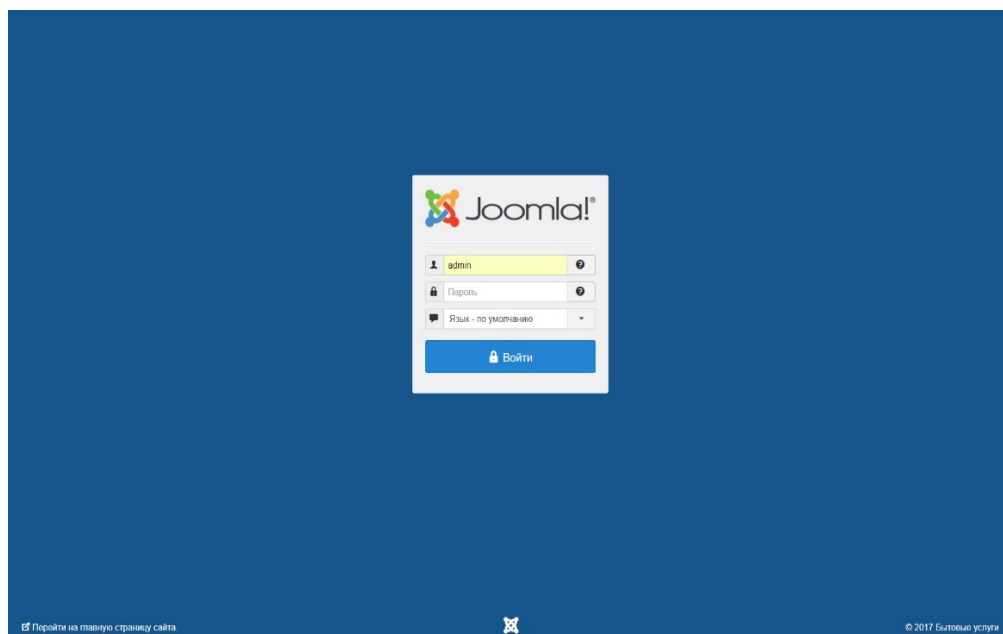


Рис. 9 Вход в административную панель

Управление учетными записями пользователей происходит через модуль «Community Builder» (Рис. 10).

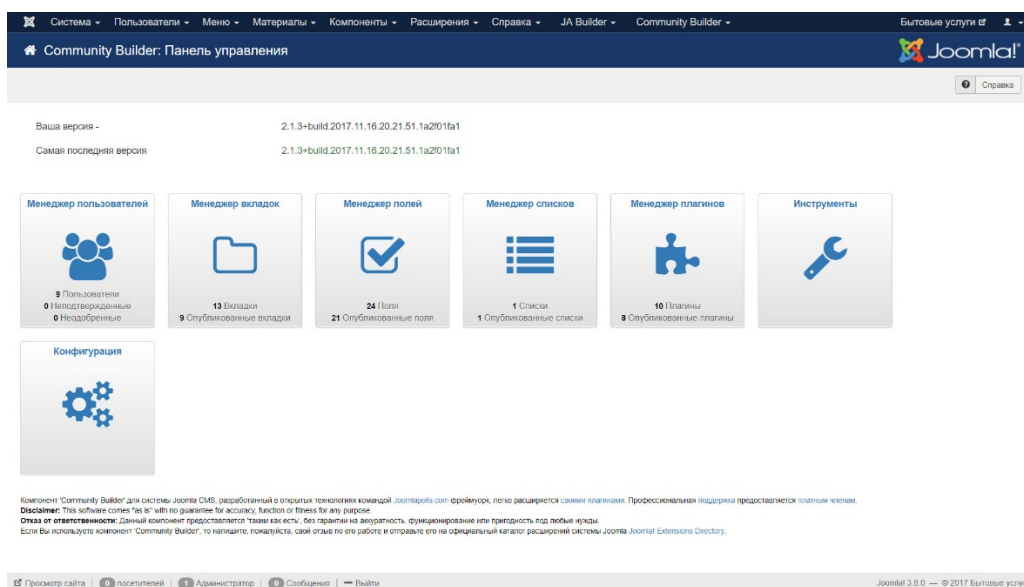


Рис. 10 Community Builder

Чтобы активировать учетную запись зарегистрированного пользователя зайдите в «Community Builder», «Менеджер пользователей», в категориях: включен, подтвержден, одобрен поставьте галочку. Также здесь имеется возможность удалять, блокировать и редактировать учетные записи пользователей (Рис. 11).

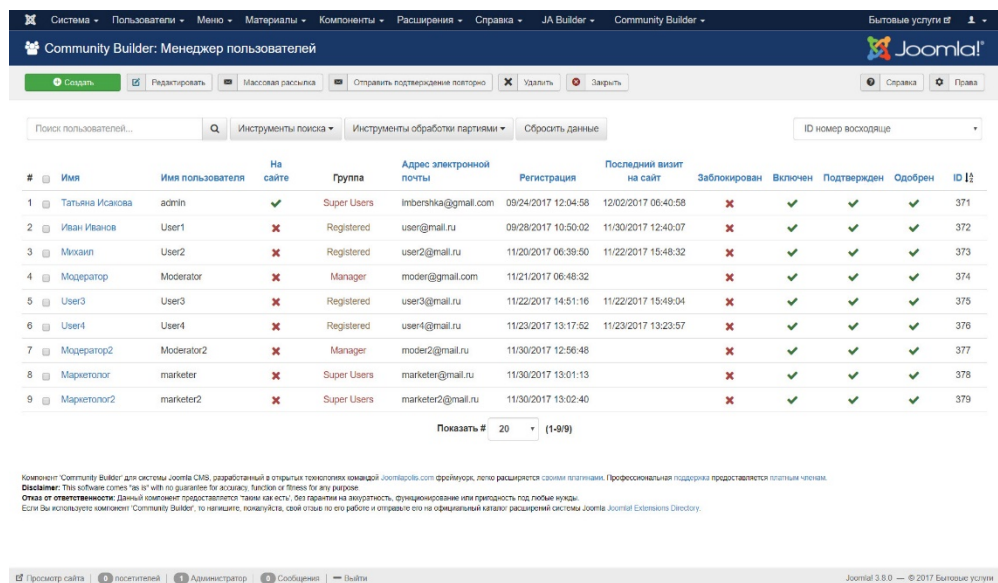


Рис. 11 Community Builder – менеджер пользователей

Все заявки от пользователей приходят на электронную почту Администратора, адрес электронной почты указывается в «Система», «Общие настройки», «Сервер» в категории «Настройка почты» (Рис. 12).

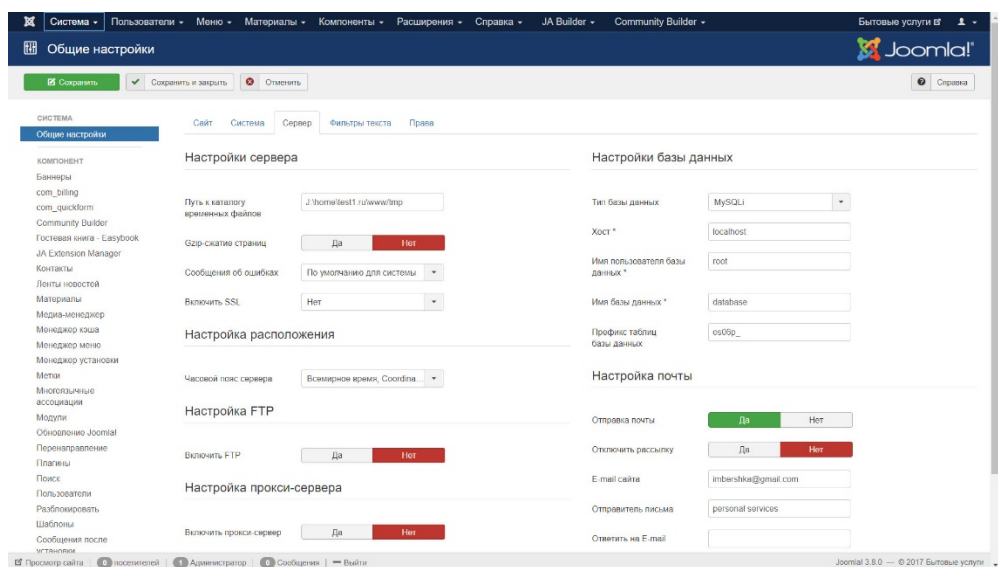


Рис. 12 Настройка почты

Заявки для исполнителей добавляются и редактируются в модуле «Quick-Form 2». Перейдите во вкладку «Компоненты», «QuickForm 2» (Рис. 13).

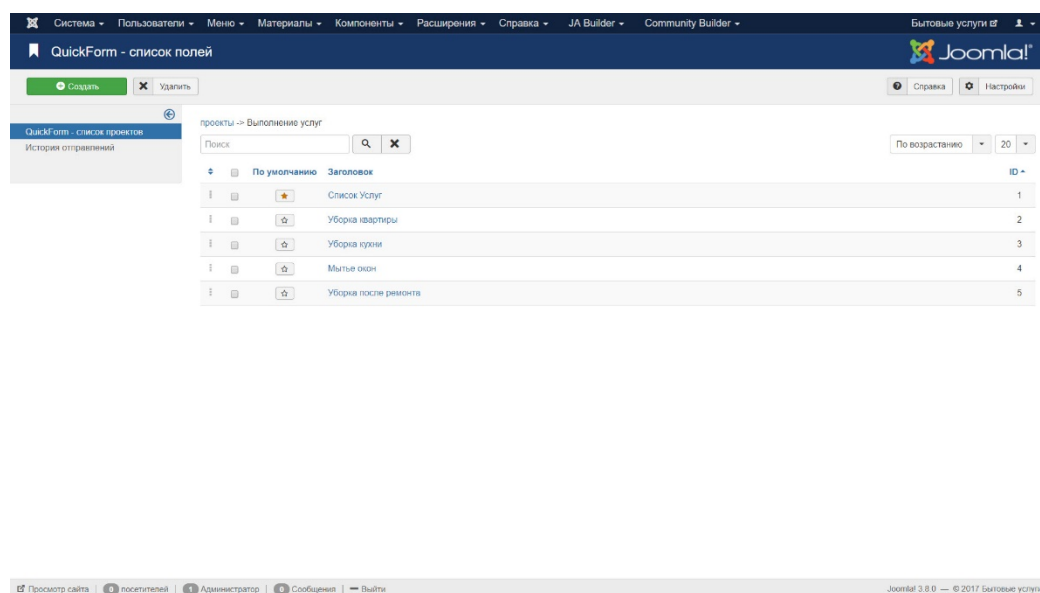


Рис. 13 Quick Form

Выберите определенную услугу (Рис. 14) для её редактирования. Или создайте новую, кнопка «Создать» (Рис. 13).

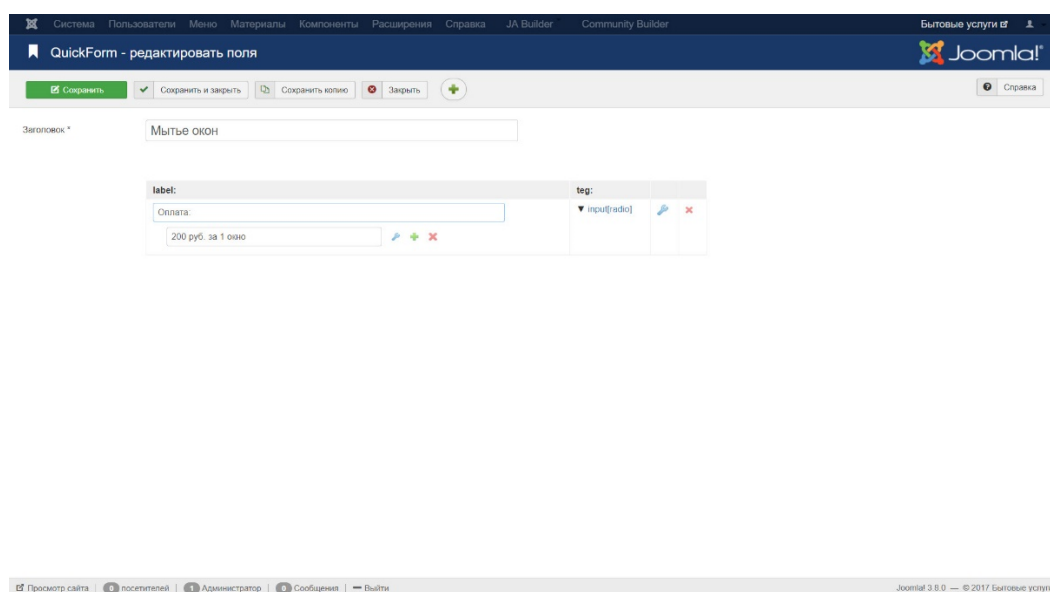


Рис. 14 Quick Form – услуги

К «Списку услуг» привязываются конкретные услуги по «ID», у каждой услуги свой ID (Рис 15).

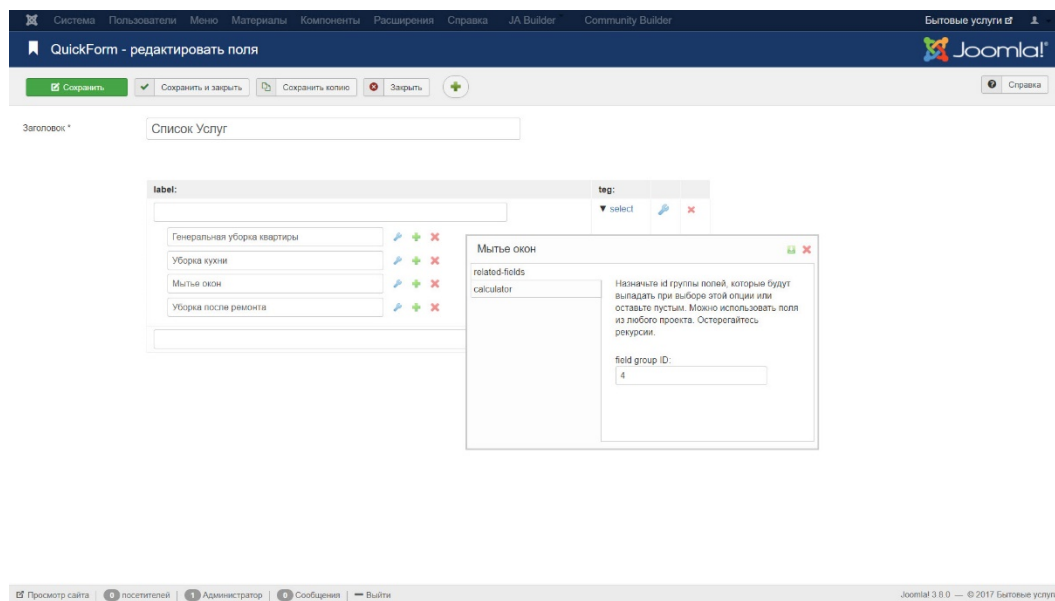


Рис. 15 Quick Form – редактирование списка услуг

Опрос «web-сервис для оказания бытовых услуг»

- 1) Как вы оцениваете внешний вид web-сервиса?
 - Отличный
 - Нормальный
 - Неприемлемый
- 2) Как вы оцениваете работу web-сервиса по шкале от 1 до 10?
- 3) Какой вид сервиса вы предпочли бы использовать?
 - Заказ услуг
 - Выполнение услуг
 - Заказ и выполнение услуг
 - Никакой
- 4) Стоит ли дополнить перечень услуг на нашем web-сервисе? Если да, то перечислите какие услуги стоит добавить в перечень. Если нет, напишите почему не стоит.
- 5) Стоит ли добавить в сервис дополнительные опции и функции, если да то какие?
- 6) Являетесь ли вы специалистом в сфере IT-технологий?
 - Являюсь
 - Нет, не являюсь
 - Интересуюсь IT-технологиями
- 7) Использовали ли вы подобный web-сервис ранее?
 - Да
 - Нет
- 8) Будете ли вы использовать данный сервис?
 - Да
 - Нет
 - Возможно